

De-Shuang Huang Donald C. Wunsch II
Daniel S. Levine Kang-Hyun Jo (Eds.)

LNAI 5227

Advanced Intelligent Computing Theories and Applications

With Aspects of Artificial Intelligence

4th International Conference on Intelligent Computing, ICIC 2008
Shanghai, China, September 2008
Proceedings

 Springer

Lecture Notes in Artificial Intelligence 5227

Edited by R. Goebel, J. Siekmann, and W. Wahlster

Subseries of Lecture Notes in Computer Science

De-Shuang Huang Donald C. Wunsch II
Daniel S. Levine Kang-Hyun Jo (Eds.)

Advanced Intelligent Computing Theories and Applications

With Aspects of Artificial Intelligence

4th International Conference
on Intelligent Computing, ICIC 2008
Shanghai, China, September 15-18, 2008
Proceedings

Series Editors

Randy Goebel, University of Alberta, Edmonton, Canada
Jörg Siekmann, University of Saarland, Saarbrücken, Germany
Wolfgang Wahlster, DFKI and University of Saarland, Saarbrücken, Germany

Volume Editors

De-Shuang Huang
Institute of Intelligent Machines
Intelligent Computing Laboratory
Chinese Academy of Sciences
Hefei, Anhui 230031, China
E-mail: dshuang@iim.ac.cn

Donald C. Wunsch II
Missouri University of Science & Technology
Department of Electrical and Computer Engineering
Applied Computational Intelligence Laboratory
Rolla, MO 65409-0040, USA
E-mail: wunsch@ieee.org

Daniel S. Levine
University of Texas at Arlington
Department of Psychology
Arlington, TX 76019-0528, USA
E-mail: levine@uta.edu

Kang-Hyun Jo
University of Ulsan
Graduate School of Electrical Engineering
Ulsan 680-749, South Korea
E-mail: jkh2008@islab.ulsan.ac.kr

Library of Congress Control Number: 2008933865
CR Subject Classification (1998): I.2.3, I.2, F.4.1, F.1, I.5, F.2, G.2, I.4
LNCS Sublibrary: SL 7 – Artificial Intelligence

ISSN 0302-9743
ISBN-10 3-540-85983-7 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-85983-3 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media

springer.com

© Springer-Verlag Berlin Heidelberg 2008
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12512720 06/3180 5 4 3 2 1 0

Preface

The International Conference on Intelligent Computing (ICIC) was formed to provide an annual forum dedicated to the emerging and challenging topics in artificial intelligence, machine learning, bioinformatics, and computational biology, etc. It aims to bring together researchers and practitioners from both academia and industry to share ideas, problems and solutions related to the multifaceted aspects of intelligent computing.

ICIC 2008, held in Shanghai, China, September 15–18, 2008, constituted the 4th International Conference on Intelligent Computing. It built upon the success of ICIC 2007, ICIC 2006 and ICIC 2005 held in Qingdao, Kunming and Hefei, China, 2007, 2006 and 2005, respectively.

This year, the conference concentrated mainly on the theories and methodologies as well as the emerging applications of intelligent computing. Its aim was to unify the picture of contemporary intelligent computing techniques as an integral concept that highlights the trends in advanced computational intelligence and bridges theoretical research with applications. Therefore, the theme for this conference was “Emerging Intelligent Computing Technology and Applications”. Papers focusing on this theme were solicited, addressing theories, methodologies, and applications in science and technology.

ICIC 2008 received 2336 submissions from 31 countries and regions. All papers went through a rigorous peer review procedure and each paper received at least three review reports. Based on the review reports, the Program Committee finally selected 401 high-quality papers for presentation at ICIC 2008, of which 373 papers have been included in three volumes of proceedings published by Springer comprising one volume of *Lecture Notes in Computer Science* (LNCS), one volume of *Lecture Notes in Artificial Intelligence* (LNAI), and one volume of *Communications in Computer and Information Science* (CCIS). The other 28 papers will be included in two international journals.

This volume of the *Lecture Notes in Artificial Intelligence* (LNAI) series includes 148 papers.

The organizers of ICIC 2008, the including the Center for International Scientific Exchanges of the Chinese Academy of Sciences, Shanghai University, and the Institute of Intelligent Machines of the Chinese Academy of Sciences, made an enormous effort to ensure the success of ICIC 2008. We hereby would like to thank the members of the ICIC 2008 Advisory Committee for their guidance and advice, and the members of the Program Committee and the referees for their collective effort in reviewing and soliciting the papers. We would like to thank Alfred Hofmann, Executive Editor at Springer, for his frank and helpful advice and guidance throughout and for his support in publishing the proceedings. In particular, we would like to thank all the authors for contributing their papers. Without the high-quality submissions from the authors, the success of the conference would not have

been possible. Finally, we are especially grateful to the IEEE Computational Intelligence Society, the International Neural Network Society and the National Science Foundation of China for their sponsorship.

July 2008

De-Shuang Huang
Donald Wunsch
Daniel S. Levine
Kang-Hyun Jo

Organization

General Chair	Donald Wunsch, USA
Steering Committee Chair	De-Shuang Huang, China
Program Committee Chair	Daniel S. Levine, USA
Organizing Committee Co-chairs	Min-Rui Fei, China
	Shi-Wei Ma, China
	Chun-Hou Zheng, China
	Ji-Xiang Du, China
Award Committee Chair	Laurent Heutte, France
Publication Chair	Kang-Hyun Jo, Korea
Special Session Chair	Marco Loog, Denmark
Tutorial Co-chairs	Fakhri Karray, Canada
	Prashan Premaratne, Australia
International Liaison Chair	Frank Neumann, Germany
Publicity Co-chairs	Vitoantonio Bevilacqua, Italy
	Wanquan Liu, Australia
	Sanggil Kang, Korea
	Plamen Angelov, UK
	Xin Li, China
Exhibition Chair	Si-Liang Chen, China

Steering Committee Members

Luonan Chen, Japan	Laurent Heutte, France	Kang Li, UK
Marco Loog, Denmark	Guangrong Ji, China	Kang-Hyun Jo, Korea
Jun Zhang, China	Xiao-Ping Zhang, Canada	

Organizing Committee Members

Jian Fan, China	Zhi-Hua Li, China	Li-Xiong Li, China
Qun Niu, China	Yang Song, China	Xin Sun, China
Ling Wang, China	Yu-Lin Xu, China	Bang-Hua Yang, China

Program Committee Members

Khalid Mahmood Aamir, Pakistan	Uwe Aickelin, UK	Vasily Aristarkhov, Russian Federation
Andrea Francesco Abate, Italy	Adel M. Alimi, Tunisia	Costin Badica, Romania
Shafayat Abrar, UK	Peter Andras, UK	Vitoantonio Bevilacqua, Italy
	Plamen Angelov, UK	
	Sabri Arik, Turkey	

- Salim Bouzerdoum, Australia
 Martin Brown, UK
 Jinde Cao, China
 Uday K., USA
 Pei-Chann Chang, Taiwan
 Peng Chen, China
 Shyi-Ming Chen, Taiwan
 Shih-Hsin Chen, Taiwan
 Weidong Chen, China
 Wen-Sheng Chen, China
 Xiyuan Chen, China
 Yuehui Chen, China
 Min-Sen Chiu, Singapore
 Michal Choras, Poland
 Tommy Chow, Hong Kong
 Jose Alfredo F. Costa, Brazil
 Kevin Curran, UK
 Mingcong Deng, Japan
 Gabriella Dellino, Italy
 Salvatore Distefano, Italy
 Ji-Xiang Du, China
 Meng Joo Er, Singapore
 Karim Faez, Iran
 Jianbo Fan, China
 Minrui fei, Canada
 Wai-Keung Fung, Canada
 Max H. Garzon, USA
 Liang Gao, China
 Ping Guo, China
 Qing-Wei Gao, China
 Xiao-Zhi Gao, Finland
 Chandan Giri, India
 Kayhan Gulez, Turkey
 Fei Han, China
 Kyungsook Han, Korea
 Aili Han, China
 Jim Harkin, UK
 Haibo He, USA
 Francisco Herrera, Spain
 Laurent Heutte, France
 Wei-Chiang Hong, Taiwan
 Yuexian Hou, China
- Guang-Bin Huang, Singapore
 Peter Chi Fai Hung, Ireland
 Won Joo Hwang, Korea
 Estevam Rafael Hruschka J., Brazil
 Myong K. Jeong, USA
 Guangrong Ji, China
 Zhenran Jiang, China
 Kang-Hyun Jo, Korea
 Jih-Gau Juang, Taiwan
 Dah-Jing Jwo, Taiwan
 Janusz Kacprzyk, Poland
 Visakan Fakhri Karray, Canada
 Hirotaka Inoue Kure, Japan
 Jia Li, China
 Kadirkamanathan, UK
 Hee-Jun Kang, Korea
 Sanggil Kang, Korea
 Uzay Kaymak, Netherlands
 Ziad Kobti, Canada
 Mario Koeppen, Japan
 Muhammad Khurram Khan, Pakistan
 Donald H. Kraft, USA
 Harshit Kumar, Korea
 Takashi Kuremoto, Japan
 Hak-Keung Lam, UK
 Sungshin Kim, Korea
 In-Soo Koo, Korea
 Yoshinori Kuno, Japan
 Turgay Ibrikci, Turkey
 Richard Lathrop, USA
 Choong Ho Lee, Korea
 Vincent C.S. Lee, Australia
 Dalong Li, USA
 Guo-Zheng Li, China
 Peihua Li, China
 Xiaoli Li, China
 Xin Li, China
 Xueling Li, China
- Hualou Liang, USA
 Chunmei Liu, USA
 Ju Liu, China
 Van-Tsai Liu, Taiwan
 Wanquan Liu, Australia
 Yanzhang Liu, China
 Ahmad Lotfi, UK
 Hongtao Lu, China
 Jinwen Ma, China
 Shiwei Ma, China
 Hiroshi Mamitsuka, Japan
 Filippo Menolascina, Italy
 Tarik Veli Mumcu, Turkey
 Roman Neruda, Czech Republic
 Frank Neumann, Germany
 Minh Nhut Nguyen, Singapore
 Ngoc Thanh Nguyen, Poland
 Sim-Heng Ong, Singapore
 Francesco Pappalardo, Italy
 Sung-Joon Park, Korea
 Daniel Patino, Argentina
 Girijesh Prasad, UK
 Prashan Premaratne, Australia
 Nini Rao, China
 Miguel Alberto Melgarejo Rey, Colombia
 Peter Rockett, UK
 Fariba Salehi, Iran
 Angel Sappa, Spain
 Karadeniz, Turkey
 Aamir Shahzad, Pakistan
 Li Shang, China
 Nobutaka Shimada, Japan
 Jiatao Song, China
 Anantaporn Srisawat, Thailand
 Nuanwan
 Soonthornphisaj, Thailand
 Joao Miguel da Costa Sousa, Portugal

Min Su, USA	Hong Wei, UK	Jun Zhang, China
Zhan-Li Sun, Singapore	Zhi Wei, China	Xi-Wen Zhang, China
Maolin Tang, Australia	Ling-Yun Wu, China	Hongyong Zhao, China
Antonios Tsourdos, UK	Shunren Xia, China	Xiaoguang Zhao, China
Naoyuki Tsuruta, Japan	Yu Xue, China	Zhongming Zhao, USA
Athanasios Vasilakos, Greece	Ching-Nung Yang, Taiwan	Bo-Jin Zheng, China
Anhua Wan, China	Jun-Heng Yeh, Taiwan	Fengfeng Zhou, USA
Chao-Xue Wang, China	Myeong-Jae Yi, Korea	Byoung-Tak Zhang, Korea
Jeen-Shing Wang, Taiwan	Xinge You, China	Xing-Ming Zhao, Japan
Jiang-Qing Wang, China	Tina Yu, Canada	Chun-Hou Zheng, China
Yong Wang, Japan	Zhi-Gang Zeng, China	Daqi Zhu, China
Zhi Wang, China	Guisheng Zhai, Japan	Xiaojin Zhu, China

Reviewers

Rahat Abbas, Janos Abonyi, Giuseppe M.C. Acciani, Ali Ahmed Adam, Alimi Adel, Muhammad Zubair Afzal, H. Agaenina, Hassan Aghaenina, Ali Aghagolzadeh, Chang Wook Ahn, Lifeng Ai, Ayca Gokhan Ak, Waseem Akhtar, Mustafa Aktas, Songul Albayrak, Davide Alemani, Rahim Ali, Ibrahim Aliskan, Muhammad Alkarouri, Abdullah Al-Malaise, Rui Jorge Almeida, Khareem Almo, Dario Aloise, Pablo Javier Alsina, Roberto T. Alves, Saleh Aly, Marco Alzate, Hamidreza Amindavar, Plamen Angelov, Dennis Barrios Aranibar, Nestor Arana Arexolaleiba, Salvatore Arinisi, Vasily Aristarkhov, Ali Ashraf-Modarres, Krassimir Atanassov, Mutlu Avci, Phillipa Avery, Erel Avineri, Thouraya Ayedi, Pedro Paulo Ayrosa, Amelia Badica, Hyeon Bae, Aditya Bagchi, Chenggang Bai, Meng Bai, Amar Balla, Lucia Ballerini, Rajib Bandyopadhyay, Sudhir Kumar Barai, Peter Baranyi, Nicola Barbarini, Jose Joel Gonzalez Barbosa, Andres Eduardo Gaona Barrera, Guilherme Barreto, Lucia Barron, Ying L. Becker, Nur Bekiroglu, Ammar Belatreche, Domenico Bellomo, Umesh Bellur, Tomas Beran, Saul Bertuccio, Alvaro Betancourt, Vitoantonio Bevilacqua, Fiachra Mac Giolla Bhríde, M.R. Bhujade, Rongfang Bie, Gennaro Nicola Bifulco, Laurentiu Biscu, P.K. Biswas, Santosh Biswas, Antonino Biundo, Dario de Blasiis, S.M. Bohte, Danail Bonchev, Andreia G. Bonfante, Olaf Booij, Giuseppe Borzi, Janez Brank, Agostinho de Medeiros Brito Junior, Dimo Brockhoff, Dario Bruneo, Ni Bu, Mari Angelica Camargo-Brunetto, Louis-Claude Canon, Galip Cansever, Anne Magali de Paula Canuto, Jianting Cao, Jinde Cao, Yang Cao, Yuan Cao, Lucia Cariello, Leonarda Carnimeo, Bianca di Angeli Carreras Simoes Costa, Bruno Motta de Carvalho, Matthew Casey, Ssa Giovanna Castellano, Marcello Castellano, Filippo Castiglione, Oscar Castillo, Pablo de Castro Roberto Catanuto, Zhiwei Cen, Jes de Jesus Fiais Cerqueira, Mark Chadwick, P. P. Chakrabarty, Mandira Chakraborty, Sandipan Chakpoborty, Chien-lung Chan, Chuan-Yu Chang, Yeong-Chan Chang, Dong Eui Chang, Kuei-Hsiang Chao, Kuei-Hsiang Chao, Liheng Chao, Hassan Tariq Chattha, Santanu Chattopadhyay, Rizwan Chaudhry, Saurabh Chaudhury, Dongsheng Che, Jiuhua Chen, Chun-Hao Chen, Cycer Chen, Chuyao Chen, Dan Chen, Shi-Jay Chen, Dongsheng Chen, Ziyi Chen, Feng-Chi Chen, Tin-Chih Chen, Yen-Ping Chen, Xuedong Chen, Zhi-Jie Chen, GS Chen, Li-Wen Chen, Miller Chen, Xinkai Chen,

Xinyu Chen, Peter Chen, Sheng Chen, Zehua Chen, Gang Chen, Ming Chen, Peng Chen, Yong Chen, Hui Chen, Ken Chen, Lin Chen, Qisong Chen, Yiming Chen, Qiming Cheng, Ming-Yang Cheng, Mu-Huo Cheng, Victor Cherepanov, Ching-Tsan Cheung, Chi Chiu Chiang, Jen-Chieh Chiang, Jerry Chien, C. H. Chin, Chaotang Chiu, Chih-Hui Chiu, Min-Sen Chiu, Leszek Chmielewski, Dong-Yeon Cho, Chang-Sik Choi, Sungsoo Choi, Sungsoo Choi, Won Ho Choi, Michal Choras, Smitashree Choudhary, Yun-Kung Chung, Andrzej Cichocki, Vincent Cicirello, Alessandro Cincotti, Guilherme Coelho, Leandro Coelho, Dorian Cojocar, Joan Condell, Oscar Cordon, Luciano da Fontoura Costa, Jose Alfredo, F. Costa, Mirel Cosulschi, Deborah Cravalho, Valentin Cristea, Cuco Cristiano, Jie Cui, Feipeng Da, Keshav Dahal, Zhifeng Dai, Hong-Yi Dai, Domenico Daleno, Nabanita Das, Bijan Davvaz, Kaushik Deb, Jayanta Kumar Debnath, Alberto Del, Bimbo Haibo Deng, Glad Deschrijver, Michael Dewar, Sajal Dey, Habib Dhahri, Jianli Ding, Alessia D'Introno, Banu Diri, Salvatore Distefano, Adriana Dobriceanu, Wenyong Dong, Yan Dong, Guy Drouin, Yongping Du, Xin Du, Mojie Duan, Fuqing Duan, Yunsuo Duan, Li Duan, Wieslaw A. Dudek, Martyn Durrant, Nees Jan van Eck, John Economou Shinto Eguchi, Chen Ei, Mehmet Kubilay Eker, Atilla Elçi, Meisam Emamjome, Seref N. Engin, Tolga Ensari, Zeki Erdem, Koksal Erenturk, Kadir Erkan, Osman Erol, Andrés Escobar, Imen Essafi, Charles Eugene, Eugene C. Ezin, Mehdi Ezoji, Umar Faiz, Alexandre Xavier Falcão, Ivanoe De Falco, Chun-I Fan, Chin yuan Fan, Shaojing Fan, Jian Fan, Xiang Fan, Kai Fan, Ping-An Fang, Yong Fang, Yi Fang, Adel Omran Farag, Sheyla Farias, Maria Fazio, Joseana Macedo Fechine, Jun Fei, Balazs Feil, Naizhang Feng, Jan Feyereisl, Sevan Ficici, Juan Carlos Figueroa, Simone Fiori, Robert Fisher, Kenneth Ford, Girolamo Fornarelli, Carlos Henrique Forster, Flavius Frasinca, Chaojin Fu, Shengli Fu, Hong Fu, Yu Fu, John Fulcher, Wai-keung Fung, Colin Fyfe, Sebastian Galvao, Zhaohui Gan, zunhai Gao, Jianxin Gao, Xiao-Zhi Gao, Qingwei Gao, Shouwei Gao, Tiehong Gao, Haibin Gao, Xin Gao, Andres Gaona, Juan Carlos Figueroa García, Alexandru Gartner, Vicente Zarzoso Gascon-Pelegri, António Gaspar-Cunha, Dingfei Ge, Fei Ge, Pando Georgiev, David Geronim, Adam Ghandar, Arfan Ghani, Pradip Ghanty, Hassan Ghasemian, Supratip Ghose, R. K. Ghosh, Marco Giannini Gustavo, Gimenez Mark Girolami, Adrian Giurca, Brendan Glackin, Cornelius Glackin, Amin Yazdanpanah Goharizi, Jackson Gomes, Márcio Leandro Gonçalves, Feng Gong, Xing Gong, Xiujun Gong, Adilson Gonzaga, Flavius Gorgonio, Diganata Goswami, Victor Hugo Grisales, André Grüning, Feng Gu, Ricardo Ribeiro Gudwin, Andrea Guerriero, Jie Gui Kayhan Gülez, Kayhan Gulez, Ge Guo, Feng-Biao Guo, Lanshen Guo, Tiantai Guo, Weiping Guo, Zheng Guo, A K Gupta, A. Gupta, Indranil Gupta, Dan Gusfield, Giménez-Lugo Gustavo, Taeho Ha, Javad Haddadnia, Tarek M. Hamdani, Yousaf Hamza, A. Han, Kyungsook Han, Kuk-Hyun Han, Lianyi Han, Kijun Han, Santoso Handri, yuanling Hao, Edda Happ, Jim Harkin, Pitoyo Hartono, Nada Hashmi, Mark Hatcher, Jean-Bernard Hayet, Guoliang He, Zhaoshui He, Zhongkun He, Zhiyong He, Hanlin He, Jun He, Liu He, Yu He, Martin Hermanto, Emilio Del Moral Hernandez, Carlos Herrera, Christian W. Hesse, Hidehiro Ohki Hidehiro, John Ho, Murillo Rodrigo Petrucelli Homem, Murillo Homem, Wei-Chiang Hong, Dihui Hong, Xia Hong, Gen Hori, Keiichi Horio, Shijinn Horng, Christian Horoba, Alamgir Hossain, Yuexian Hou, Zhixiang Hou, Guolian Hou, Estevam R. Hruschka Jr., Chen-Huei Hsieh, Jih-Chang Hsieh, Jui-chien Hsieh, Sun-Yuan Hsieh, Chi-I Hsu, Yu-Liang Hsu, Dan Hu, Yongqiang Hu, Xiaolin

Hu, Ting Hu, YAN Hua, Chuanxiu Huang, Jian Huang, Wei-Hsiu Huang, Sun-Jen Huang, Weichun Huang, Weitong Huang, Ying J. Huang, Yuefei Huang, Jian Huang, Ping Huang, Di Huang, Evan J Hughes, Yung-Yao Hung, Changyue Huo, Knut Huper, Saiful Huq, Kao-Shing Hwang, I-Shyan Hwang, Won-Joo Hwang, Mintae Hwang, Hwang, Wonju Hwang, Muhammad Usman Ilyas, Anca Ion, Ahmad Ali Iqbal, Zahid Irfan, Y. Ishida, Ivan Nunes Silva, Kuncup Iswandy, Marcin Iwanowski, Yumi Iwashita, Sridhar Iyer, Gonçalves, J. F., Beirão, N., Saurabh Jain, Lakhmi Jain, Sanjay Kumar Jana, D. Janakiram, Jun-Su Jang, Marko Jankovic, Mun-Ho Jeong, Zhi-Liang Ji, Hongjun Jia, Wei Jia, Jigui Jian, Cizhong Jiang, Chang-An Jiang, Yuncheng Jiang, Minghui Jiang, Xingyan Jiang, Lihua iang, Bin Jiao, Kyohong Jin, Zhong Jin, Rong Jin, Geunsik Jo, Jang Wu Jo, Torres-Sospedra Joaquin, Daniel Johannsen, Colin Johnson, José Demisio Simões da Silva, R.K. Joshi, Tejal Joshi, Koo Joungsun, Jih-Gau Juang, Carme Julià, Young Bae Jun, Heesung Jun, Khurum Nazir Junejo, Jinguk Jung, Francisco Madeiro Bernardino Junior, Roberto Marcondes Cesar Junior, Dah-Jing Jwo, Osvaldo Mafra Lopes Junio, E. Kabir, Visakan Kadirkamanathan, Salim Kahveci, kaka, Ilhem Kallel, Habib Kammoun, Hamid Reza Rashidy Kanan, Hyunduk Kang, Hyun-Deok Kang, Hee-June Kang, Hyunduk Kang, Henry Kang, Yasuki Kansha, Cihan Karakuzu, Ghader Karimian, Bekir Karlik, Shohreh Kasaei, Faisal M Kashif, Boer-Sorbán Katalin, H Kawasaki, Olesya Kazakova, Christel Kemke, Tamas Kenesei, Selami Kesler, Muhammad Khurram Khan, Malik Jahan Khan, Shehroz Khan, Pabitra Mohan Khilar, Pabitra Khilar, Chin Su Kim, Chungsan Kim, Dae-Nyeon Kim, Myung-Kyun Kim, Kane Kim, Pil Gyeom Kim, Seong Joo Kim, Eunchan Kim, Gwan-Su Kim, Hak Lae Kim, Kanghee Kim, Il Kon Kim, Sung S Kim, Taeho Kim, Christian Klein, Chun-Hsu Ko, Yoshinori Kobayashi, Kunikazu Kobayashi, Andreas Koenig, Mario Koeppen, Andrew Koh, xiangzhen Kong, Insoo Koo, Murakami Kouji, Vladik Kreinovich, Ibrahim Kucukdemiral, Rajeev Kumar, Chao-Lin Kuo, Tzu-Wen Kuo, Wen-Chung Kuo, Simon Kuo, Takashi Kuremoto, Zarei-Nia Kurosh, Janset Kuvulmaz, Yung-Keun Kwon, Chien-Yuan Lai, Franklin Lam, H.K. Lam, Andrey Larionov, Pietro Larizza, M. Mircea Lazar, Vincenzo Di Lecce, Yulia Ledeneva, Bore-Kuen Lee, Chiho Lee, Kyung Chang Lee, Vincent C S Lee, Myung-Joon Lee, Guanling Lee, Hong-Hee Lee, Ka-keung Lee, Shao-Lun Lee, Eun-Mi Lee, In-Hee Lee, Sangho Lee, Minhoo Lee, N.-Y. Lee, Peter Lee, Lee, Lee, Suwon Lee, Vincent Lee, Per Kristian Lehre, Yujun Leng, Agustin Leon, Carson K. Leung, Alexandre Levada, Ao Li, Caiwei Li, Chen Li, Chia-Hsiang Li, Chien-Kuo Li, Bo Li, Mingdong Li, Hualiang Li, Weigang Li, KeQing Li, Xinyu Li, Heng-Chao Li, Guozheng Li, Hongchun Li, Kangshun Li, Qingfeng Li, Xiaodong Li, zhisheng Li, HuiFang Li, Renwang Li, Shanbin Li, Xueling Li, Yueping Li, Liyuan Li, Rewang Li, Shutao Li, Yiyang Li, Fuhai Li, Li Erguo, Jian Li, Yong Li, Lei Li, Min Li, Feng-Li Lian, Yun-Chia Liang, Hualou Liang, Han Liang, Liao, Wudai Liao, Hee-Woong Lim, Cheng-Jian Lin, Chih-Min Lin, Feng-Yan Lin, Jyun Jie Lin, Jyun-Yu Lin, Jun-Lin Lin, Yu-Chen Lin, Jimmy Lin, Lin, Hao Lin, Junjie Lin, Yingbiao Ling, Steve Ling, Chang Liu, Che-Wei Liu, Bingqiang Liu, Yubao Liu, Xingcheng Liu, Yongmei liu, Jing Liu, Mei-qin Liu, Qingshan Liu, Van-Tsai Liu, KunHong Liu, liangxu liu, Shiping Liu, Weiling Liu, Xiaomin Liu, Xiaoyue Liu, Yu-ling Liu, Zhiping Liu, Hongbo Liu, Jizhen Liu, Liu, Yifan Liu, Qian Liu, Xiao Liu, Jin Liu, Jun Liu, Yue Liu, Joe K. W. Lo, Asim Loan, Andrey Logvinov, Francesco Longo, Milan Lovric, Baoliang Lu, Yixiang Lu, Junguo

Lu, Feng Lu, June Lu, Wei Lu, CJ Luh, Luiz Marcos Garcia Gonçalves, Andrew Lumsdaine, Tom Lunney, Jingchu Luo, Yan Luo, Leh Luoh, Yan Lv, Chuang Ma, Yinglong Ma, Liyong Ma, Irwin Ma, Jin Ma, Sakashi Maeda, Sakashi Maeda, Sudipta Mahapatra, Sydulu Maheswarapu, Andre Laurindo Maitelli, A.K. Majumdar, Chandan Majumdar, Terrence Mak, Hiroshi Mamitsuka, Qing-Kui Man, Achintya Kumar Mandal, Danilo Mandic, Mata-Montero ManriqueAtif Mansoor, Chengxiong Mao, Zhiming Mao, Fenglou Mao, Zhihong Mao, Weihua Mao, Kezhi Mao, Joao Fernando Marar, Márcio Leandro Gonçalves Mario Marinelli, Francescomaria Marino Urszula Markowska-Kaczmar, Alan Marshall, Allan de Medeiros Martins, Nelson Delfino d Avila Mascarenhas, Emilio Mastriani, Giuseppe Mastronardi, Francesco Masulli, Mohammad Ali Maud, Giancarlo Mauri, Joseph McClay, Liam McDaid, Malachy McElholm, Adelardo A. Dantas de Medeiros, Claudio Medeiros, Reginald Mehta, Jorge Dantas de Melo, Luis Mendonca, Weixiao Meng, Filippo Menolascina, Jianxun Mi, Hirvensalo Mika, Nikolay Mikhaylov, Claudia Milaré, Viorel Milea, Milos Radovanovic, Mihoko Minami, Tsunenori Mine, Giuseppe Minutoli, Sushmita Mitra, Mandar Mitra, Yasue Mitsukura, Jinqiu Mo, Asunción Mochón, Hamid Abrishami, Moghaddam Hamid, Abrishami Moghaddam, Nurul Haque Mollah, Marina Mongiello, Inhyuk Moon, Fearghal Morgan, Yasamin Mostofi, Santo Motta, saeed Mozaffari, Mikhail Mozerov, Krishnendu Mukhopadhyay, J. Mukhopadhyay, Hamid Mukhtar, Tarik Veli Mumcu, T. Murakami, C. Siva Ram Murthy, Muhammad Aziz Muslim, Kazuo Nakamura, Sukumar Nandi, David Naso, Pedro L.K.G Navarro, Duarte Dória Neto, Frank Neumann, WK Ng, Hoi Shing Raymond NG, Tian-Tsong Ng, Vinh Hao Nguyen, Tam Nguyen, Ni, Oana Nicolae, Li Nie, Ke Ning, Luis F. Nino, Fauzia Nisar, Maria Nisar, Takuichi Nishimura, Qun Niu, Shimada Nobutaka, Lars Nolle, Clement Nyirenda, Masanao Obayashi, Hasan Ocak, Richard Oentaryo, Jaewon Oh, Halil Ibrahim Okumus, M. Sorin Olaru, Luiz Affonso H Guedes de Oliveira, Pietro Oliveto, Onat, Kok-Leong Ong, Johan Oppen, Denis Orel, Ajiboye Osunleke, Gaoxiang Ouyang, Ali Ozen, Opraog Pag, Umapada Pal, luca Paladina, Sarbani Palit, Shanliang Pan, Tianhong Pan, Wan-Ling Pan, Paolo Pannarale, Maurizio Paone, Angelo Paradiso, Emerson Paraiso, Daniel Paraschiv, Sang Kyeong Park, Jintae Park, Swapan Kumar Parui, Halit Pastaci, Giuseppe Patanè, Athanasios Pavlou, Jeronimo Pellegrini, Jeronimo Pellegrini, Wei Peng, Marzio Pennisi, Graziano Pesole, Emil Petre, Alfredo Petrosino, Minh-Tri Pham, Vinhthuy Phan, Francesco Piazzaa, Aderson Pifer, Pinar, Huseyin Polat, Alexander Ponomarenko, Alisa Ponomarenko, Elvira Popescu, Girijesh Prasad, Prashan Premaratne, Adam Prugel_bennett, Andrzej Przybyszewski, Viswanath Pulabaigari, Alfredo Pulvirenti, Liu Qian, Haiyan Qiao, Lishan Qiao, Yu Qiao, Hong Qin, Jun Qin, Ying-qiang Qiu, ying qiu, Dong-Cai Qu, Tho Quan, Paulo Quintiliano, Ijaz Mansoor Qureshi, Tariq Rasheed Qureshi, Anas Quteishat, S.V. Raghavan, Carmelo Ragusa, Mkm Rahman, Anca Ralescu, Ramon Zatarain-Cabada, Milton Ramos, Zeeshan Rana, Raquel Esperanza Patiño Escarcina, Jiangtao Ren, Jian Ren, Alberto Rey, Orion Fausto Reyes-Galaviz, Robert Reynolds, Gianbattista Rocco, Peter Rockett, Liu Rong, A.K. Roy, Kaushik Roy, Uttam Roy, Changhai Ru, XiaoGang Ruan, Tomasz Rutkowski, Khalid Saeed, Doris Sáez, Alaa Sagheer, G. Saha, Ratnesh Sahay, Halil Ibrahim Sahin, Mohamed Sahmoudi, G Sajith, Pijush Samui, Saeid Sanei, David Sankoff, Edimilson B. dos Santos, Jose Santos, Brahmananda Sapkota, Angel Sappa, P.Saratchandran, Yoshiko Sato, Gerald

Schaefer, Giuseppe Scionti, Dan Selisteanu, S. Selvakumar, Kirusnapillai Selvarajah, Amitava Sen, Sibel Senan, Dorin Sendrescu, Indranil Sengupta, D.Y. Sha, A Shah, Syed Faisal Ali Shah, Syed Ismail Shah, Suleman Shahid, Bilal Shams, Shahnoor Shanta, Li Shao, Qadeer Sharif, Shahzad Amin Sheikh, Hao Shen, Xianjun Shen, Yantao Shen, Yehu Shen, Jinn-Jong Sheu, Chuan Shi, MingGuang Shi, Yongren Shi, Ke Shi, Horng-Lin Shieh, Motoki Shiga, Atsushi Shimada, Tetsuya Shimamura, Soo-Yong Shin, Wochang Shin, Tae zi Shin, Takahashi Shinozaki, Dipak Lal Shrestha, Bi Shuhui, Leandro Augusto da Silva, Fulvio Simonelli, Leszek Sliwko, Kate A.Smith, Grant Smith, Heliana B. Soares, Zhuoyue Song, Qiankun Song, Yinglei Song, Ong Yew Soon, Nuanwan Soonthornphisaj, Jairo Soriano, Joao M. C. Sousa, Marcilio Carlos P. de Souto, Jackson Gomes de Souza, Birol Soysal, Stefano Squartini, Mscislaw Srutek, Cristina Stoica, Umberto Straccia, Antony Streklas, Zheng Su, Min Su, Ahlada Sudersan, Akira Sukanuma, Youngsoo Suh, Ziwen Sun, Tsung-Ying Sun, Tien-Lung Sun, Xiangyang Sun, Jingchun Sun, Shiwei Sun, Lily Sun, Yude Sun, Nak Woon Sung, Seokjin Sung, Worasait Suwannik, Aqeel Syed, Duong Ta, Abdullah Taha, Chen Tai, Oluwafemi Taiwo, Shin-ya Takahashi, B. Talukdar, Hakaru Tamukoh, Guangzheng Tan, Ping Tan, Toshihisa Tanaka, Chunming Tang, Hong Tang, David Taniar, Zou Tao, Liang Tao, Imran Tasadduq, Peter Tawdross, Mohammad Teshnehlab, Niwat Thepvilojanapong, Daniel Thiele, Quan Thanh Tho, Jingwen Tian, Jiang Tian, Yun Tian, Ye Tian, Huaglory Tianfield, Ching-Jung Ting, Massimo Tistarelli, Stefania Tommasi, Ximo Torres, Farzad Towhidkhal, Cong Tran-Xuan, Roque Mendes Prado Trindade, Hoang-Hon Trinh, Gianluca Triolo, Giuseppe Troccoli, Chieh-Yuan Tsai, Chi-Yang Tsai, Chueh-Yung Tsao, Norimichi Tsumura, Naoyuki Tsuruta, Hang Tu, Hung-Yi Tu, Luong Trung Tuan, Petr Tuma, Cigdem Turhan, Francesco Tusa, Bulent Tutmez, Seiichi Uchida, Muhammad Muneeb Ullah, Nurettin Umurkan, Mustafa Unel, Ray Urzulak, Ernesto Cuadros Vargas, Andrey Vavilin, Simona Venuti, Silvano Vergura, Susana Vieira, Geoffrey Vilcot, Massimo Villari, Boris Vintimilla, Holger Voos, Juan Wachs, John Wade, Hiroshi Wakuya, Julie Wall, Li Wan, Bohyeon Wang, Chao Wang, Chengyou Wang, Xingce Wang, Jia-hai Wang, Jiasong Wang, Guoli Wang, Yadong Wang, Xiaomin Wang, Jeen-Shing Wang, Zhongsheng Wang, Guoren Wang, Xiangyang Wang, Zhongxian Wang, Jianying Wang, LingLing Wang, Ruisheng Wang, Xiaodong Wang, XiaoFeng Wang, Xiaojuan Wang, Xiaoling Wang, Xuan Wang, Zhengyou Wang, Haijing Wang, Hesheng Wang, Hongxia Wang, Hongyan Wang, Jianmin Wang, Junfeng Wang, Linshan Wang, Shuting Wang, Yanning Wang, Zhisong Wang, Huimin Wang, Huisen Wang, Mingyi Wang, Shulin Wang, Zheyou Wang, Haili Wang, Jiang Wang, Kejun Wang, Linze Wang, Weiwu Wang, Jina Wang, Jing Wang, Ling Wang, Meng Wang, Qifu Wang, Yong Wang, Yan Wang, Yoshikazu Washizawa, Shih-Yung Wei, Shengjun Wen, Shenjun Wen, Guozhu Wen, Seok Woo, Derek Woods, Chao Wu, Christine Wu, Zikai Wu, Hsiao-Chun Wu, Quanjun Wu, YongWei Wu, Ing-Chyuan Wu, Shiow-yang Wu, Shiqian Wu, Shaochuan Wu, Wen-Chuan Wu, JianWu Wu, Weimin Wu, Qiong Wu, Sitao Wu, Peng Wu, Min Wu, Jun-Feng Xia, Li Xia, Yongkang Xiao, Jing Xiao, Lijuan Xiao, Renbin Xiao, Gongnan Xie, Zhijun Xie, Caihua Xiong, Wei Xiong, ChunGui Xu, Chunsui Xu, Weidong Xu, Wenlong Xu, Xiaoyin Xu, Zeshui Xu, Huan Xu, Wei Xu, Yun Xu, Xuanli Wu, Quan Xue, Yu Xue, Xuesong Yan, Li Yan, Banghua Yang, Junghua Yang, Wuchuan Yang, Yingyun Yang, Hyunho Yang, Junan Yang, Shixi

Yang, Sihai Yang, Song Yang, Yan Yang, Ming-Jong Yao, Xingzhong Yao, Daoxin Yao, Obilor Yau, Xiaoping Ye, Liang Ye, Chia-Hsuan Yeh, Ming-Feng Yeh, Jun-Heng Yeh, James Yeh, Yang Yi, Tulay Yildirim, Jian Yin, Zhouping Yin, Qian Yin, Yang Yong, Murilo Lacerda Yoshida, Norihiko Yoshida, Kaori Yoshida, Kenji Yoshimura, Mingyu You, Yu Sun Young, Changrui Yu, Gwo-Ruey Yu, Xinguo Yu, Ming Yu, Tina Yu, Zhiyong Yuan, Guili Yuan, Fang Yuan, Jing Yuan, Jing Yuan, Eylem Yucel, Lu Yue, Masahiro Yukawa, Mi-ran Yun, C. Yung, Anders Zachrison, Aamer Zaheer, Kun Zan, Yossi Zana, Rafal Zdunek, Zhigang Zeng, Wenyi Zeng, Chuan-Min Zhai, Byoung-Tak Zhang, Chuan Zhang, Dabin Zhang, Guangwei Zhang, Ping Zhang, Xianxia Zhang, Yongmin Zhang, Xiangliang Zhang, Zhiguo Zhang, Jingliang Zhang, De-xiang Zhang, Xiaowei Zhang, Xiaoxuan Zhang, Yongping Zhang, Jianhua Zhang, Junpeng Zhang, Shanwen Zhang, Si-Ying Zhang, Weigang Zhang, Yonghui Zhang, Zanchao Zhang, Zhiyong Zhang, Guohui Zhang, Guowei Zhang, Jiakai Zhang, Li-bao Zhang, Liqing Zhang, Yunong Zhang, Zhijia Zhang, LiBao Zhang, Wenbo Zhang, Jian Zhang, Ming Zhang, Peng Zhang, Ping Zhang, Zhen Zhang, Fei Zhang, Jie Zhang, Jun Zhang, Li Zhang, Bo Zhao, Xiaoguang Zhao, Quanming Zhao, Xiaodong Zhao, Yinggang Zhao, Zengshun Zhao, Yanfei Zhao, Ting Zhao, Yaou Zhao, Qin Zhao, Xin Zhao, Yi Zhao, Bojin Zheng, Xin Zheng, Yi Zheng, Aimin Zhou, Chi Zhou, Chunlai Zhou, Xiaocong Zhou, Fengfeng Zhou, Qinghua Zhou, Jiayin Zhou, Zekui Zhou, Qiang Zhou, Wei Zhou, Dao Zhou, Hao Zhou, Jin Zhou, Wen Zhou, Zhongjie Zhu, Quanmin Zhu, Wei Zhu, Hankz Zhuo, Majid Ziaratban.

Table of Contents

Neural Networks

A Miniature Robot System with Fuzzy Wavelet Basis Neural Network Controller	1
<i>Lianzhi Yu, Guozheng Yan, Chunyang Wang, Wenlong Yang, Peng Zan, and Jie Wu</i>	
A New Algorithm for Finding the Shortest Path Tree Using Competitive Pulse Coupled Neural Network	9
<i>Dongming Zhou, Rencan Nie, and Dongfeng Zhao</i>	
Criteria for Exponential Stability of Cohen-Grossberg Neural Networks with Multiple Time-Varying Delays	19
<i>Anhua Wan and Weihua Mao</i>	
Detecting Unsuccessful Automated Negotiation Threads When Opponents Employ Hybrid Strategies	27
<i>Ioannis Papaioannou, Ioanna Roussaki, and Miltiades Anagnostou</i>	
Electromagnetism-Like Mechanism Based Algorithm for Neural Network Training	40
<i>Xiao-Juan Wang, Liang Gao, and Chao-Yong Zhang</i>	
Evolvable Neuro-fuzzy System for Artificial Creativity in Linguistics	46
<i>Keith Douglas Stuart and Maciej Majewski</i>	
Extended Legendre Wavelets Neural Network	54
<i>XiaoYang Zheng, XiaoFan Yang, Hong Su, and Yong Wu</i>	
Growing Algorithm of Laguerre Orthogonal Basis Neural Network with Weights Directly Determined	60
<i>Yunong Zhang, Tongke Zhong, Wei Li, Xiuchun Xiao, and Chenfu Yi</i>	
MATLAB Simulation and Comparison of Zhang Neural Network and Gradient Neural Network for Online Solution of Linear Time-Varying Matrix Equation $AXB - C = 0$	68
<i>Ke Chen, Shuai Yue, and Yunong Zhang</i>	
Motion Detection Using Spiking Neural Network Model	76
<i>QingXiang Wu, T.M. McGinnity, Liam Maguire, Jianyong Cai, and G.D. Valderrama-Gonzalez</i>	

On Determination of Early Warning Grade Based on AHP Analysis in Warranty Database 84
SangHyun Lee, MinTae Lee, Culhyun Kim, ChulSu Park, SeungBeom Park, Yuyang Liu, and ByungKi Kim

Optimizing the Performance of Probabilistic Neural Networks Using PSO in the Task of Traffic Sign Recognition 90
LunBo Li and GuangFu Ma

Product Schemes Evaluation Method Based on Improved BP Neural Network 99
Weiwei Chen, Xiaopeng Wei, and Tingting Zhao

Study on Tool Wear Monitoring Based on Multi-source Information Fusion 107
Lanshen Guo, Haiwei Zhang, Yanxia Qi, and Zhi Wei

Synchronization and Parameter Identification for a Class of Chaotic Neural Networks with Time-Varying Delays Via Adaptive Control 115
Zhongsheng Wang, Yanjun Liang, and Nin Yan

Evolutionary Computing and Genetic Algorithms

A Discrete Binary Version of the Electromagnetism-Like Heuristic for Solving Traveling Salesman Problem 123
Nikbakhsh Javadian, Mohsen Gol Alikhani, and Reza Tavakkoli-Moghaddam

A Frequent Pattern Mining Algorithm for Understanding Genetic Algorithms 131
Minh Nghia Le and Yew Soon Ong

A Generalized Differential Evolution Combined with EDA for Multi-objective Optimization Problems 140
Wang Chen, Yan-jun Shi, and Hong-fei Teng

A New Hybrid Ant Colony Optimization Algorithm for the Traveling Salesman Problem 148
Xiaoxia Zhang and Lixin Tang

A Novel PSO-DE-Based Hybrid Algorithm for Global Optimization 156
Ben Niu and Li Li

A Parallel Multi-algorithm Solver for Dynamic Multi-Objective TSP (DMO-TSP) 164
Ming Yang, Zhou Kang, and Lishan Kang

A Virtual Three-Dimension Cellular Automata Pseudorandom Number Generator Based on the Moore Neighborhood Method	174
<i>Sang-Ho Shin, Geum-Dal Park, and Kee-Young Yoo</i>	
An Adaptive Genetic Algorithm for Solving Traveling Salesman Problem	182
<i>Jina Wang, Jian Huang, Shuqin Rao, Shaoe Xue, and Jian Yin</i>	
An Application of Support Vector Machines for Induction Motor Fault Diagnosis with Using Genetic Algorithm	190
<i>Ngoc-Tu Nguyen and Hong-Hee Lee</i>	
Ant Algorithm Applied in the Minimal Cost Maximum Flow Problem	201
<i>Min Xie, Lixin Gao, and Haiwa Guan</i>	
Computer Simulation and Word-Updating Dynamical Systems (WDS) on Digraphs	209
<i>Deqiang Chen, Jie Zheng, and Xiaoqian Wu</i>	
Cooperative Dynamics in Spatially Structured Populations	217
<i>Lihui Shang</i>	
Emergent Motion Characteristics of a Modular Robot through Genetic Algorithm	225
<i>Sunil Pranit Lal, Koji Yamada, and Satoshi Endo</i>	
Evolutionary Optimization with Dynamic Fidelity Computational Models	235
<i>Dudy Lim, Yew-Soon Ong, Yaochu Jin, and Bernhard Sendhoff</i>	
Grid Resource Aggregation Integrated P2P Mode	243
<i>Zenggang Xiong, Yang Yang, Xuemin Zhang, and Ming Zeng</i>	
Hybrid Scatter Search with Extremal Optimization for Solving the Capacitated Vehicle Routing Problem	251
<i>Kai Sun, Gen-Ke Yang, and Yu-Wang Chen</i>	
Implement Web Learning System Based on Genetic Algorithm and Pervasive Agent Ontology	259
<i>Qinglin Guo and Ming Zhang</i>	
Large Population Size IGAs with Individuals' Fitness Not Assigned by User	267
<i>Jie Yuan and Dunwei Gong</i>	
Missing Data Imputation in Time Series by Evolutionary Algorithms	275
<i>Juan C. Figueroa García, Dusko Kalenatic, and Cesar Amilcar Lopez Bello</i>	

Rule-Based Analysis of Behaviour Learned by Evolutionary and Reinforcement Algorithms	284
<i>Stanislav Šlušný, Roman Neruda, and Petra Vidnerová</i>	
Self-Guided Genetic Algorithm	292
<i>Shih-Hsin Chen, Pei-Chann Chang, and Qingfu Zhang</i>	
Study on Discretization in Rough Set Via Modified Quantum Genetic Algorithm	300
<i>Shuhong Chen and Xiaofeng Yuan</i>	
Fuzzy Systems and Soft Computing	
A Class of Random Fuzzy Programming and Its Hybrid PSO Algorithm	308
<i>Yankui Liu, Xuejie Bai, and Fang-Fang Hao</i>	
A Fuzzy Logic Approach to Test Statistical Hypothesis on Means	316
<i>Juan Carlos Figueroa García and Jose Jairo Soriano Mendez</i>	
A Fuzzy Logic-Based Approach to Two-Dimensional Warranty System	326
<i>SangHyun Lee, DongSu Lee, ChulSu Park, JaeHee Lee, SeungBeom Park, KyungIl Moon, and ByungKi Kim</i>	
A Fuzzy Optimization Method for Multi-criteria Decision-Making Problem Based on the Inclusion Degrees of Intuitionistic Fuzzy Sets	332
<i>Changrui Yu and Yan Luo</i>	
A GP Based Approach to the Classification of Multiclass Microarray Datasets	340
<i>Chun-Gui Xu and Kun-Hong Liu</i>	
A Novel Approach for Blind Channel Equalization	347
<i>A. Özen, A. Güner, O. Çakır, E. Tuğcu, B. Soysal, and I. Kaya</i>	
A Risk-Sensitive Portfolio with Mean and Variance of Fuzzy Random Variables	358
<i>Yuji Yoshida</i>	
Attribute Reduction Based on the Fuzzy Information Filter Operators	367
<i>Fachao Li, Chao Gao, and Chenxia Jin</i>	
Classification of Image Data Using Gradient-Based Fuzzy C-Means with Mercer Kernel	376
<i>Dong-Chul Park</i>	

Design of T-S Fuzzy Model Based on PSODE Algorithm	384
<i>Ben Niu and Li Li</i>	
Equivalence Knowledge Mass and Approximate Reasoning in \mathcal{R} -Logic $\mathbb{C}_{\mathcal{R}}$ (I)	391
<i>Yalin Zheng, Guang Yang, Changshui Zhang, Jing Zheng, and Yunpeng Xu</i>	
Extended Fuzzy Knowledge Representation with Medium	401
<i>Cen Wang and Zhenghua Pan</i>	
Five Perspectives on Case Based Reasoning	410
<i>Zhaohao Sun, Jun Han, and Dong Dong</i>	
Fixpoint Semantics and Completeness of the Computational Model for Fuzzy Linguistic Logic Programming	420
<i>Van Hung Le, Fei Liu, and Dinh Khang Tran</i>	
Forest-RK: A New Random Forest Induction Method	430
<i>Simon Bernard, Laurent Heutte, and Sébastien Adam</i>	
Fuzzy Linguistic Logic Programming	438
<i>Van Hung Le, Fei Liu, and Dinh Khang Tran</i>	
Improve Flow Accuracy and Byte Accuracy in Network Traffic Classification	449
<i>Haitao He, Chunhui Che, Feiteng Ma, Xiaonan Luo, and Jianmin Wang</i>	
Intuitionistic Fuzzy Modules and Their Structures	459
<i>Chuanyu Xu</i>	
Optimization of Ni-MH Battery Fast Charging in Electric Vehicles Using Dynamic Data Mining and ANFIS	468
<i>Guifang Guo, Peng Xu, Zhifeng Bai, Shiqiong Zhou, Gang Xu, and Binggang Cao</i>	
Robust and Adaptive Fuzzy Feedback Linearization Regulator Design	476
<i>Chang-Woo Park, Young-Ouk Kim, and Chong Ho Yi</i>	
Three-Level Image Segmentation Based on Maximum Fuzzy Partition Entropy of 2-D Histogram and Quantum Genetic Algorithm	484
<i>Hai-Yan Yu and Jiu-Lun Fan</i>	
Particle Swarm Optimization and Niche Technology	
A Hybrid Particle Swarm Optimization for Feed-Forward Neural Network Training	494
<i>Ben Niu and Li Li</i>	

A Modified Ant Colony Algorithm Used for Multi-robot Odor Source Localization 502
Yuhua Zou and Dehan Luo

A New Problem-Solving Method for Multi-agent Systems and Computer Networks: Solitary Wave Propagation 510
Dianxun Shuai

A Novel Global Convergence Algorithm: Bee Collecting Pollen Algorithm 518
Xueyan Lu and Yongquan Zhou

An Enhanced Swarm Intelligence Clustering-Based RBF Neural Network Detection Classifier 526
Yong Feng, Zhong-fu Wu, Jiang Zhong, Chun-xiao Ye, and Kai-gui Wu

Application of the Particle Filter for Simple Gesture Recognition 534
Yang Weon Lee

Cooperative Approaches to Bacterial Foraging Optimization 541
Hanning Chen, Yunlong Zhu, Kunyuan Hu, Xiaoxian He, and Ben Niu

Crossbar Composite Spring-Nets to Optimize Multi-Agent Systems and Computer Networks 549
Dianxun Shuai

Formation Control of Multiple Robots Using Parametric and Implicit Representations 558
Yesim H. Esin, Mustafa Unel, and Mehmet Yildiz

Fuzzy SVM Training Based on the Improved Particle Swarm Optimization 566
Ying Li, Bendu Bai, and Yanning Zhang

Handling Multiobjective Problems with a Novel Interactive Multi-Swarm PSO 575
Yujia Wang and Yupu Yang

The Explicit Exploration Information Exchange Mechanism for Niche Technique 583
Jun Zhang and Kwok-Wing Chau

The Research of Using Ant Colony Algorithm in Solving Sequencing Problem of Mixed Model Assembly Lines with Multi-objectives 591
Qiong Zhu, Lihui Wu, and Jie Zhang

Supervised and Semi-supervised Learning

A New Weighted Support Vector Machine for Regression and Its Parameters Optimization 597
Liquan Mei and Shujuan Zhang

A Novel Spike Sorting Method Based on Semi-supervised Learning	605
<i>Guo-Zhu Wen and De-Shuang Huang</i>	
A One-Step Network Traffic Prediction	616
<i>Xiangyang Mu, Nan Tang, Weixin Gao, Lin Li, and Yatong Zhou</i>	
A Two-Step Selective Region Ensemble for Facial Age Estimation	622
<i>Shenglan Ben, Guangda Su, and Youshou Wu</i>	
A Unified String Kernel for Biology Sequence	633
<i>Dehui Yuan, Shengyun Yang, and Guoming Lai</i>	
An Adaptively Constructing Multilayer Feedforward Neural Networks Using Hermite Polynomials	642
<i>L. Ma and K. Khorasani</i>	
Improved Learning Algorithms of SLFN for Approximating Periodic Function	654
<i>Fei Han</i>	
Some Progress of Supervised Learning	661
<i>Chunyang Su, Shifei Ding, Weikuan Jia, Xin Wang, and Xinzheng Xu</i>	
Supervised Gravitational Clustering with Bipolar Fuzzification	667
<i>Umut Orhan, Mahmut Hekim, and Turgay Ibrikci</i>	

Unsupervised and Reinforcement Learning

A Neuro-fuzzy Learning System for Adaptive Swarm Behaviors Dealing with Continuous State Space	675
<i>Takashi Kuremoto, Masanao Obayashi, Kunikazu Kobayashi, Hirotaka Adachi, and Kentaro Yoneda</i>	
A Reinforcement Learning Automata Optimization Approach for Optimum Tuning of PID Controller in AVR System	684
<i>Mohammad Kashki, Youssef Lotfy Abdel-Magid, and Mohammad Ali Abido</i>	
An Alternative to Center-Based Clustering Algorithm Via Statistical Learning Analysis	693
<i>Rui Nian, Guangrong Ji, and Michel Verleysen</i>	

Fine Feature Extraction Methods

A Survey on Statistical Pattern Feature Extraction	701
<i>Shifei Ding, Weikuan Jia, Chunyang Su, Fengxiang Jin, and Zhongzhi Shi</i>	

Nonlinear Innovation to Noisy Blind Source Separation Based on Gaussian Moments 709
Hongjuan Zhang, Chonghui Guo, Zhenwei Shi, and Enmin Feng

Removal of Artifacts in Electroencephalogram Using Adaptive Infomax Algorithm of Blind Source Separation 717
Wanyou Guo, Liyu Huang, Li Gao, Tianqiao Zhu, and Yuanqi Huang

Combinatorial and Numerical Optimization

A Hybrid VNS with TS for the Single Machine Scheduling Problem to Minimize the Sum of Weighted Tardiness of Jobs..... 727
Xianpeng Wang and Lixin Tang

A Wide Neighborhood Primal-Dual Interior-Point Algorithm for a Class of Convex Programming 734
Yuqin Zhao and Mingwang Zhang

An Improved Differential Evolution with Local Search for Constrained Layout Optimization of Satellite Module 742
Wang Chen, Yan-jun Shi, and Hong-fei Teng

Cloud Theory Based Simulated Annealing Algorithm for Multiple Observers Sitting on Terrain Problem..... 750
Pin Lv, Lin Yuan, and Jinfang Zhang

Minimum Vertex Ranking Spanning Tree Problem on Some Classes of Graphs 758
Ruei-Yuan Chang, Guanling Lee, and Sheng-Lung Peng

On Constructions of Optimal Double-Loop Networks 766
Jianqin Zhou and Wenjuan Wang

Research on Resource Selection with Precedence and Due Date Constraint 775
Shujuan Li, Yong Liu, Yan Li, Yuefei Xu, and Zhibin Zeng

Rule-Based Modeling of Assembly Constraints for Line Balancing 783
Latif Salim and Aliye Ayca Supciller

Neural Computing and Optimization

A Novel Network Behavior Model Based on Generalized Cellular Automaton..... 790
Dianxun Shuai

Optimal Control of Switched System Based on Neural Network Optimization	799
<i>Rong Long, Jinming Fu, and Liyan Zhang</i>	
Zhang Neural Network Versus Gradient Neural Network for Online Time-Varying Quadratic Function Minimization	807
<i>Yunong Zhang, Zhan Li, Chenfu Yi, and Ke Chen</i>	
Case Based Reasoning and Autonomy-Oriented Computing	
A Logic Description on Different Negation Relation in Knowledge	815
<i>Zhenghua Pan</i>	
An On-Line Semantic Visualization for Heterogeneous Product Information of the Virtual Organization	824
<i>Chengfeng Jian</i>	
An Optimized Parallel Decision Tree Model Based on Rough Set Theory	832
<i>Xiaowang Ye and Zhijing Liu</i>	
A Topic-Specific Web Crawler with Concept Similarity Context Graph Based on FCA	840
<i>Yuekui Yang, Yajun Du, Jingyu Sun, and Yufeng Hai</i>	
An IACO and HPSO Method for Spatial Clustering with Obstacles Constraints	848
<i>Xueping Zhang, Jiayao Wang, Dexian Zhang, and Zhongshan Fan</i>	
Attributes Reduct and Optimal Decision Rules Acquisition in Fuzzy Objective Information Systems	857
<i>Fang Yang, Yan-Yong Guan, and Zhao-Xia Yu</i>	
Cross Sentence Alignment for Structurally Dissimilar Corpus Based on Singular Value Decomposition	864
<i>Anna Ho, Fai Wong, Francisco Oliveira, and Yiping Li</i>	
Event Detection and Tracking Based on Improved Incremental K-Means and Transductive SVM	872
<i>Zhen Lei, Jianfeng Liao, Dong Li, and Lingda Wu</i>	
Integration of Fuzzy Logic in Data Mining to Handle Vagueness and Uncertainty	880
<i>G. Raju, Binu Thomas, Th. Shanta Kumar, and Sangay Thinley</i>	
Medical Knowledge Discovery from a Regional Asthma Dataset	888
<i>Sam Schmidt, Gang Li, and Yi-Ping Phoebe Chen</i>	

Mining Context-Specific Web Knowledge: An Experimental Dictionary-Based Approach	896
<i>Vincenzo Di Lecce, Marco Calabrese, and Domenico Soldo</i>	
Mining Direct and Indirect Fuzzy Multiple Level Sequential Patterns in Large Transaction Databases	906
<i>Weimin Ouyang, Qinhua Huang, and Shuanghu Luo</i>	
Rough Set Approaches for Mining Incomplete Information Systems	914
<i>M.K. Sabu and G. Raju</i>	
Φ -Rough Sets Theory and Its Usage on Mining Approximate Dependencies	922
<i>Yiyong Xiao, Ikou Kaku, and Wenbing Chang</i>	

Artificial Life and Artificial Immune Systems

A Cellular Automatic Method for the Edge Detection of Images	935
<i>Yu Chen and Zhuangzhi Yan</i>	
Computer Virus Evolution Model Inspired by Biological DNA	943
<i>Yu Zhang, Tao Li, and Renchao Qin</i>	
Study of Immune Control Computing in Immune Detection Algorithm for Information Security	951
<i>Ya Jing Zhang and Yang Xue</i>	

Special Session on Computational Analysis and Data Mining in Biological Systems

A Comparative Study of Machine Learning Methods for Detecting Promoters in Bacterial DNA Sequences	959
<i>Leonardo G. Tavares, Heitor S. Lopes, and Carlos R. Erig Lima</i>	
A Holistic View of Evolutionary Rates in Paralogous and Orthologous Genes	967
<i>Deng Pan and Liqing Zhang</i>	
An SVM-Based Algorithm for Classifying Promoter-Associated CpG Islands in the Human and Mouse Genomes	975
<i>Leng Han, Ruolin Yang, Bing Su, and Zhongming Zhao</i>	
CDGMiner: A New Tool for the Identification of Disease Genes by Text Mining and Functional Similarity Analysis	982
<i>Fang Yuan and Yanhong Zhou</i>	
Designing Genetic Regulatory Network with Fuzzy Logic	990
<i>Jianwei Shen, Yi Wang, and Zengrong Liu</i>	

A Framework to Understand the Mechanism of Toxicity	996
<i>Wenhui Huang and Liqing Zhang</i>	
FindSUMO: A PSSM-Based Method for Sumoylation Site Prediction . . .	1004
<i>Christopher J. Friedline, Xueping Zhang, Zendra E. Zehner, and Zhongming Zhao</i>	
Maximal-Robustness-Minimal-Fragility Controller: A Compromise between Robustness and Fragility of Biochemical Networks	1012
<i>Ming-Guang Shi, Michael R. Lyu, and Tat-Ming Lok</i>	
Predict Molecular Regulatory Network of Norway Rat under the Frame of Data Integration	1022
<i>Qiguo Rong and Zhe Zhang</i>	
Using Cost-Sensitive Learning to Determine Gene Conversions	1030
<i>Mark J. Lawson, Lenwood Heath, Naren Ramakrishnan, and Liqing Zhang</i>	

Special Session on Data Mining and Fusion in Bioinformatics – Advances in Soft Computing Approach

A Multi-objective Genetic Algorithm Based Approach to the Optimization of Oligonucleotide Microarray Production Process	1039
<i>Filippo Menolascina, Vitoantonio Bevilacqua, Caterina Ciminelli, Mario Nicola Armenise, and Giuseppe Mastronardi</i>	
A New Orthogonal Discriminant Projection Based Prediction Method for Bioinformatic Data	1047
<i>Chao Wang and Bo Li</i>	
A Novel Hybrid Method of Gene Selection and Its Application on Tumor Classification	1055
<i>Zhuhong You, Shulin Wang, Jie Gui, and Shanwen Zhang</i>	
An Intelligent Analyzer for Supporting Diagnoses of Congenital CMV Infection	1069
<i>Leonarda Carnimeo</i>	
Biomedical Text Mining Using a Grid Computing Approach	1077
<i>Marcello Castellano, Giuseppe Mastronardi, Giacinto Decataldo, Luca Pisciotto, Gianfranco Tarricone, Lucia Cariello, and Vitoantonio Bevilacqua</i>	
High-Throughput Analysis of the Drug Mode of Action of PB28, MC18 and MC70, Three Cyclohexylpiperazine Derivative New Molecules	1085
<i>Vitoantonio Bevilacqua, Paolo Pannarale, Giuseppe Mastronardi, Amalia Azzariti, Stefania Tommasi, Filippo Menolascina, Francesco Iorio, Diego Di Bernardo, Angelo Paradiso, Nicola A. Colabufo, Francesco Berardi, Roberto Perrone, and Roberto Tagliaferri</i>	

Locally Linear Discriminant Embedding for Tumor Classification	1093
<i>Chun-Hou Zheng, Bo Li, Lei Zhang, and Hong-Qiang Wang</i>	
Palmprint Linear Feature Extraction and Identification Based on Ridgelet Transforms and Rough Sets	1101
<i>Shanwen Zhang, Shulin Wang, and Xuelin Li</i>	
Special Session on Intelligent Pattern Recognition and Understanding in Image Processing	
A Compact Genetic Algorithm with Elitism and Mutation Applied to Image Recognition	1109
<i>Rafael R. Silva, Heitor S. Lopes, and Carlos R. Erig Lima</i>	
A Method of Counting Pedestrians in Crowded Scenes	1117
<i>Byeoung-su Kim, Gwang-Gook Lee, Ja-Young Yoon, Jae-Jun Kim, and Whoi-Yul Kim</i>	
A Novel Approximate Algorithm Used for Solving the Netting Radar Measure Elevation Angle Based on the Target Altitude	1127
<i>Wen-Bo Zhao, Ji-Yan Du, Li-Ming Wang, and Yong-Ke Chen</i>	
Automated Segmentation Using a Fast Implementation of the Chan-Vese Models	1135
<i>Huan Xu and Xiao-Feng Wang</i>	
Automatic Facial Feature Points Detection	1142
<i>Vitoantonio Bevilacqua, Alessandro Ciccimarra, Ilenia Leone, and Giuseppe Mastronardi</i>	
Automatic Subspace Synthesis of Motion Styles Based on Isomap	1150
<i>Jian Xiang and Hongli Zhu</i>	
Card Images Binarization Based on Dual-Thresholding Identification . . .	1158
<i>Chunmei Liu, Duoqian Miao, and Chunheng Wang</i>	
Defects Identification in Textile by Means of Artificial Neural Networks	1166
<i>Vitoantonio Bevilacqua, Lucia Cariello, Giuseppe Mastronardi, Vito Palmieri, and Marco Giannini</i>	
Diabetic Damage Detection in Retinal Images Via a Sparsely-Connected Neurofuzzy Network	1175
<i>Leonarda Carnimeo</i>	
Dimensionality Reduction Method Applied for Multi-view Face Database Recognition	1183
<i>Hui-Ming Huang and Zhi-Kai Huang</i>	

Experimental System to Support Real-Time Driving Pattern Recognition	1192
<i>Vincenzo Di Lecce and Marco Calabrese</i>	
Extending Hough Transform to a Points' Cloud for 3D-Face Nose-Tip Detection	1200
<i>Vitoantonio Bevilacqua, Pasquale Casorio, and Giuseppe Mastronardi</i>	
Face Detection by Means of Skin Detection	1210
<i>Vitoantonio Bevilacqua, Giuseppe Filograno, and Giuseppe Mastronardi</i>	
Research on Product Case Representation and Retrieval Based on Ontology	1221
<i>Yan Shen, XiaoPeng Wei, and JianWei Wang</i>	
Retinal Fundus Biometric Analysis for Personal Identifications	1229
<i>Vitoantonio Bevilacqua, Lucia Cariello, Donatello Columbo, Domenico Daleno, Massimiliano Dellisanti Fabiano, Marco Giannini, Giuseppe Mastronardi, and Marcello Castellano</i>	
Statistical Character Study of the Inert Coordinate Errors Caused by the Measurement Noise of the Netting Radar	1238
<i>Wen-Bo Zhao, Xiao-Chun Mei, Chuan Li, and Yong-Ke Chen</i>	
Author Index	1247

A Miniature Robot System with Fuzzy Wavelet Basis Neural Network Controller

Lianzhi Yu¹, Guozheng Yan², Chunyang Wang¹, Wenlong Yang¹,
Peng Zan², and Jie Wu¹

¹ College of Optoelectric and Electric Information Engineering, University of Shanghai for
Science and Technology, Shanghai, 200093, P.R. China

yulianzhi001@163.com

² School of Electronic, Information and Electrical Engineering, Shanghai Jiaotong
University, Shanghai, 200240, P.R. China

gzghan@sjtu.edu.cn

Abstract. This paper described the structure of a flexible miniature robot system which can move in human cavities, it makes inchworm-like movement driven by a 3-DOF pneumatic rubber actuator and holds its positions by air chambers. The driving characteristics in axial and bending directions of the actuator were analyzed and the robot control system was designed. The four-layer neural network and the five-layer neural network controllers were proposed to estimate the straight movement and the bending movement respectively. The tracking response and the tracking errors were studied in detail. Results prove that good static and dynamic control effects of the robot system can be obtained by the proposed fuzzy wavelet basis neural networks.

1 Introduction

In recent years, with the development of micro electro mechanical systems and micro sensor techniques, the development of micro robot in medical inspection and surgery has been concerned more and more about the aged society. Generally, robots are driven by electromagnetism, piezoelectricity, special materials such as shape memory alloy etc., or pneumatic rubber actuator, and the mobile mechanisms are leg, wheel, crawler or combination of them [1-4]. A medical robot directly contacting human body requires safety and flexibility, so conventional rigid mechanisms are not suitable for medical robot in human body, the soft and squirming moving style is the best. So the robot system with flexible structure mechanism was designed, and a pneumatic rubber actuator was adopted to drive the system. The locomotion mode of system was based on the inchworm movement. Because of the compressible character of air, to obtain high accuracy control, it is necessary to analyze the characteristics of the pneumatic robot system and study the effective control algorithms. For nonlinear dynamical systems, it has been proven that artificial neural network can approximate a wide range of nonlinear functions to any desired degree of accuracy under certain conditions [5-8]. Moreover, much research has been done on applications of wavelet neural networks which combine the capability of artificial neural networks in learning from the processes and the capability of wavelet decomposition [9-11]. As fuzzy control does not rely on the object

mathematic model, it has good robust and can overcome the nonlinear and the uncertain parameters of the system [10], so by wavelet basis function as fuzzy membership function, it can adjust the shape of the membership function online in real time, the fuzzy control will has some adaptive ability by utilizing the learning ability of the artificial neural network. So in this paper, the fuzzy wavelet basis neural networks were proposed to control the pneumatic micro robot system.

2 Structure of the Miniature Robot System

The structure and the sizes of the robot system had been discussed in Refs. [12,13]. The structure of the robot system is shown in Fig.1, while the basic characteristics of the prototype are listed in Table 1.

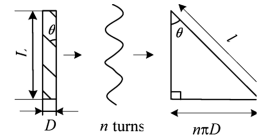
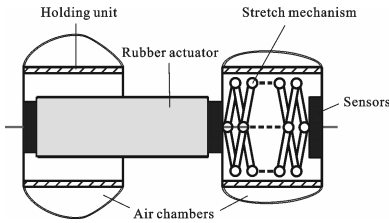


Fig. 1. Structure of the robot system

Fig. 2. Rubber actuator

Fig. 3. Rubber parameters

Table 1. Structure characteristics

Symbol	Name	Characteristics
D_0	Diameter of actuator	6 mm(o.d.), 4 mm (i.d.)
L_0	Length of actuator	20 mm
L_1	Length of front holder	10 mm
L_2	Length of rear holder	10 mm
D	Holding cylinder	12 mm(o.d.), 11 mm (i.d.)
L_3	Length of robot body	30 mm
M	Mass of robot body	about 2 g

The structure of the pneumatic robot system is composed of three parts: the front holder, the driving part and the rear holder. In moving state, the holders are used to hold the position of the system body, and the driving part is used to change the positions of the holders. The two holders are designed with cylindrical structure and covered with air chambers outside. Each holder has two interconnected air chambers; when charged, the two chambers are kept in the same pressure, and hold tightly against the inner wall. The driving part is a pneumatic rubber actuator made of fiber-reinforced rubber with three degrees of freedom. The actuator rubber tube is divided into three identical sector chambers. When the three chambers of the actuator are charged with the same air pressure, the actuator will stretch in axial direction, and when only one of them is charged, the actuator will bend to the direction opposite to the other chambers. The structure of the actuator is shown in Fig.2, and the geometrical structure parameters are shown in Fig.3.

3 Driving Characteristics of the Rubber Actuator

The driving force in axial direction of the 3-DOF pneumatic rubber actuator can be derived in the same way as Chou did for one DOF pneumatic rubber actuator according to the conservation and transformation of energy [14], the ideal model of the 3-DOF actuator driving force in axial direction had been obtained in [13]. The relation between the driving force, the pneumatic pressure and the displacement is expressed in Fig.4. The simulation results show that the maximum driving force of the actuator is over 3 N, and the maximum effective displacement is about 6 mm.

The bending characteristics of 3-DOF actuator can be described by three parameters, θ , R and λ as shown in Fig.5 [15]. When only one chamber of the actuator is charged, the action forces of the actuator are shown in Fig.6. In the view of the equivalence between the bending sectors, the angle that the actuator bending-axis projection makes with the axis x in x - y reference frame can be got [13, 15]. The theoretical deflection angle-pressure curve is a straight line, as shown in Fig.7.

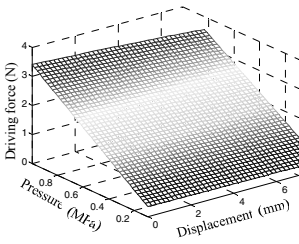


Fig. 4. Axial driving forces

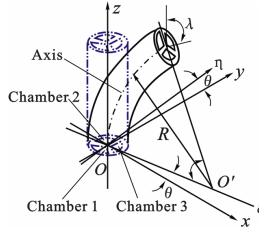


Fig. 5. Deflection angle

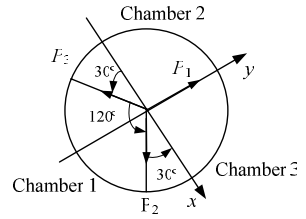


Fig. 6. Forces distribution

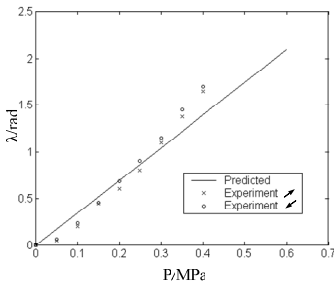


Fig. 7. Deflection angles of the actuator

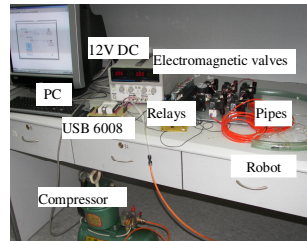


Fig. 8. Experimental control system

4 Control System of the Robot

The moving mechanism and time control orders of the robot have been described in Ref. [5]. An experimental electro-pneumatic pressure control system was designed. It mainly consists of a computer, an USB card, a compressor, ten relays, ten electromagnetism valves, pressure regulator valves and some pipes. The robot system has five pneumatic pressure pipes to be controlled. Each relay controls an electromagnetic

valve, every pressure supply pipe is controlled by a 2/2 valve and a 3/2 electromagnetic valve. The control system can control the robot moving according to its locomotion orders by LabVIEW programs. The control system is shown in Fig.8.

5 Fuzzy Wavelet Basis Neural Network Controllers

5.1 Wavelet Basis Membership Function and Wavelet Neural Network

The shape of the membership function can be adjusted online in real time when the wavelet basis function is adopted as the membership function. The wavelet basis membership function can be expressed as:

$$\mu_{A_j}^i(x_j) = h_{a_{ij}, b_{ij}}(x_j), h_{a_{ij}, b_{ij}}(x_j) = h\left[\frac{x_j - b_{ij}}{a_{ij}}\right], h(x) = \cos(0.5x) \exp(-x^2 / 2). \quad (1)$$

Where x_j is the membership degree of the language variable sets A_j^i ; $h_{a_{ij}, b_{ij}}$ is the wavelet basis function; a_{ij}, b_{ij} are the translation and dilation factors of wavelet respectively, and h is the mother wavelet. The three wavelet basis functions in $[-1, 1]$ is shown in Fig.9.

The structure of three-layer wavelet neural network is shown in Fig.10. The Mexican Hat Wavelet is used to be as activation function. In Fig.10 x_1, x_2, \dots, x_3 , is the input signals, they are pneumatic pressure errors and the change of the pressure errors; $\Psi(z_i)$ is the wavelet function; y is the output signal. The imply neuron outputs of the wavelet neural network and the output signal can be calculated as follows:

$$\Psi(z_i) = \frac{2}{\sqrt{3}} \pi^{-1/4} (1 - z_i) e^{-z_i^2/2}, z_i = \sum_{j=1}^n (x_j a_{ij} - b_i), y = \sum_{i=1}^k \omega_i \Psi(z_i). \quad (2)$$

Where $i=1,2,3 \dots k, j=1,2,3 \dots n$; ω_i is weight coefficient between the imply layer.

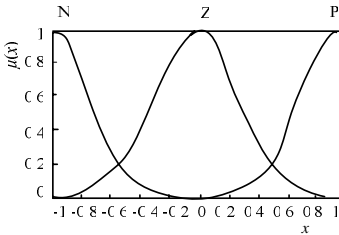


Fig. 9. Wavelet basis membership function

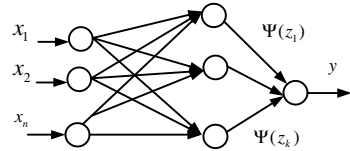


Fig. 10. Wavelet basis neural network

5.2 Fuzzy Wavelet Neural Networks for the Robot

The control objects of the robot system are two holders and a three-tube rubber actuator. So the pressures in the five supply pipes need to be controlled well. The four-layer fuzzy wavelet neural network controller was set up for the every two holder of

the robot, which is shown in Fig.11. As the three rubber tubes can not be made completely equivalent, there are also coupling actions among them actually, even when the pressures of the three tubes are different, so five-layer wavelet neural network controller is proposed for the actuator as shown in Fig.12.

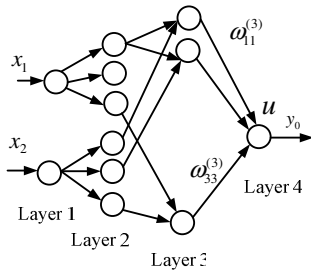


Fig. 11. Four-layer neural network

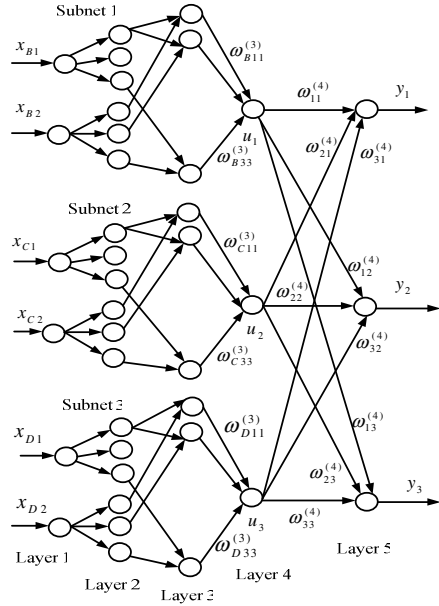


Fig. 12. Four-layer neural network

The relationships of every layer can be got. There are three same subnets B, C, and D for the every three rubber tuber of the actuator, The subnet C and subnet D are the same as the subnet B, so only the relationships of every layer for subnet B is described as following:

In the first layer, the inputs are introduced into the network with region in $[-1, 1]$, it can be expressed as:

$$out_{Bi}^{(1)} = in_{Bi}^{(1)} = x_{Bi}^{(1)}, \quad i=1,2. \quad (3)$$

The inputs are fuzzed in the second layer, the relationship of the input and output for is expressed as:

$$out_{Bij}^{(2)} = in_{Bij}^{(2)} = \mu_{Ai}^j (out_{Bi}^{(1)}), \quad i=1,2, \quad j=1, 2, 3. \quad (4)$$

The third layer is ‘and’ operation, the relationships of the input and output is expressed as:

$$out_{Bij}^{(3)} = in_{Bij}^{(3)} = out_{B1i}^{(2)} * out_{B2j}^{(2)}, \quad i=j=1, 2, 3. \quad (5)$$

The fourth layer is the operation of contrary blur, the relationship of the input and output for subnet B can be expressed as:

$$out_{Bij}^{(3)} = in_{Bij}^{(3)} = out_{B1i}^{(2)} * out_{B2j}^{(2)}, \quad u_1 = out_B^{(4)} = in_B^{(4)} / \sum_{i,j=1}^3 out_{Bij}^{(3)}. \quad (6)$$

The fifth layer stands for the coupling actions of the three rubber tubers of the actuator, which is expressed as:

$$y_j = \sum_{i=1}^3 (u_i * \omega_{ij}^{(4)}), \quad j=1, 2, 3. \quad (7)$$

5.3 Simulation Experiments

The gradient learning algorithm was used to adjust $\omega_{Bij}^{(3)}$, $\omega_{Cij}^{(3)}$, $\omega_{Dij}^{(3)}$, $\omega_{ij}^{(4)}$, and the wavelet translation factor a_{ij} , dilation factor b_{ij} , the objective function is defined as:

$$J = \sum_k \|y_{id}(k) - y_i(k)\|^2. \quad (8)$$

Where $i=0, 1, 2$, y_{id} is the expectative output, and y_i is the actual output.

The moving steps of the robot in axial direction are chosen as 1 mm, 2 mm, 3 mm, 4 mm, 5 mm, 6 mm respectively, the tracking response is shown in Fig.13, and the average tracking error is shown in Fig.14. The bending angles are chosen as 0.2 rad, 0.4 rad, 0.6 rad, 0.8 rad, 1.0 rad, 1.2 rad respectively, the tracking response is shown in Fig.15, and the average tracking error is shown in Fig.16. The simulation results indicate the controller has high speed response and less over shoot.

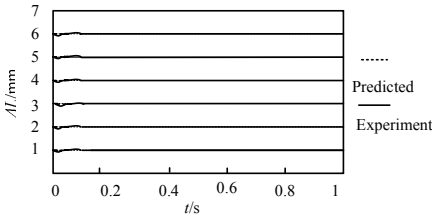


Fig. 13. Tracking response in axial direction

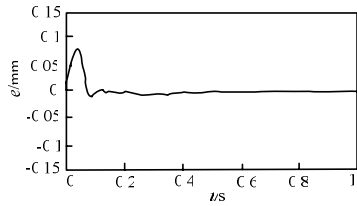


Fig. 14. Tracking error in axial direction

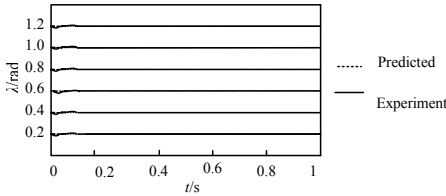


Fig. 15. Bending Tracking response

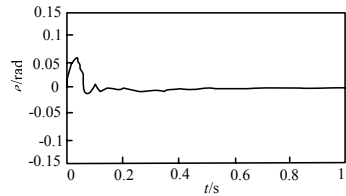


Fig. 16. Bending Tracking error

6 Conclusions

The characteristics of the pneumatic robot system had been analyzed through theory models and experiments; the robot system has enough driving force and deflection angles. The proposed fuzzy wavelet basis neural network for the robot has good static and dynamic performance, and by the fuzzy wavelet basis neural network controller, the pneumatic pressure actuating robot system can be controlled in high accuracy.

Acknowledgements. This work was supported by National High Technology Research and Development Program of China (863 Program, No.: 2004AA404013), by the PhD foundation of University of Shanghai for Science and Technology (No.: 10D209), and by National Natural Science Foundation of China (No.: 50576056).

References

1. Ikeuchi, K., Yoshinaka, K., Hashimoto, S., Tomita, N.: Locomotion of Medical Micro Robot with Spiral Ribs Using Mucus. In: The 7th IEEE International Symposium on Micro Machine and Human Science, pp. 217–222. IEEE Press, Nagoya (1996)
2. Anthierens, C., Libersa, C., Touaibia, M., Betemps, M., Arsicault, M., Chaillet, N.: Micro Robots Dedicated to Small Diameter Canalization Exploration. In: 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 480–485. IEEE Press, Kagawa University (2000)
3. Yan, G.Z., Zuo, J.Y.: A Self-propelling Endoscope System by Squirmly Robot. In: 2003 International Symposium on Micromechatronics and Human Science, pp. 159–163. IEEE Press, Nagoya (2003)
4. Thomann, M., Betemps, R.T.: The Design of a New Type of Micro Robot for the Intestinal Inspection. In: 2002 IEEE International Workshop on Robot and Human Interactive communication, pp. 1385–1390. IEEE Press, Berlin (2002)
5. Lewis, F.L., Liu, K., Yesildirek: Neural Net Robot Controller with Guaranteed Tracking Performance. *IEEE Trans. Neural Netw.* 6, 703–715 (1995)
6. Leu, Y.G., Wang, W.Y., Lee, T.T.: Observer-Based Direct Adaptive Fuzzy-Neural Control for Nonaffine Nonlinear System. *IEEE Trans. Neural Netw.* 16(4), 853–861 (2005)
7. Pati, Y.C. and Krishnaprasad, P.S.: Analysis and Synthesis of Feed-Forward Neural Networks Using Discrete Affine Wavelet Transformations. *IEEE Trans. Neural Netw.* 4, 73–85 (1993)
8. Ho, D.W.C., Li, J., Niu, Y.: Adaptive Neural Control for a Class of Nonlinearly Parametric Time Delayed System. *IEEE Trans. Neural Netw.* 16, 625–635 (2005)
9. Bilings, S.A., Wei, H.L.: A New Class of Wavelet Networks for Nonlinear System Identification. *IEEE Trans. Neural Netw.* 4, 862–874 (2005)
10. Hui, Y.L., Qiao, J.F.: Fuzzy Neural Network Control of Uncertain Parameters System. In: The 5th World Congress on Intelligent Control and Automation, Hongzhou, China, pp. 2612–2616 (2004)
11. Lin, F.J., Hsin-Jang, Huang, P.K.: Adaptive Wavelet Neural Network Control with Hysteresis Estimation for Piezo-positioning Mechanism. *IEEE Trans. Neural Netw.* 17, 432–444 (2006)
12. Yu, L.Z., Yan, G.Z., Zhang, W.Q., Wang, X.R.: Research on an Active and Continuous Monitoring System for Human Respiratory System. *High Technology Letters* 12, 68–71 (2006)

13. Yu, L.Z., Yan, G.Z., Wang, X.G.: A Soft Micro-Robot System for Direct Monitoring in Human Trachea. *Robot* 28, 269–274 (2006)
14. Chou, C.P., Hannaford, B.: Static and Dynamic Characteristics of McKibben Pneumatic Artificial Muscles. In: 1994 IEEE Robotic and Automation Conference, pp. 281–286. IEEE Press, San Diego (1994)
15. Suzumori, K., Likura, S., Tanaka, H.: Applying a Flexible Microactuator to Robotic Mechanisms. In: 1991 IEEE International Conference on Robotic and Automation, pp. 22–27. IEEE Press, Sacramento (1991)

A New Algorithm for Finding the Shortest Path Tree Using Competitive Pulse Coupled Neural Network

Dongming Zhou, Rencan Nie, and Dongfeng Zhao

Information College, Yunnan University, Kunming, 650091, China
zhoudm@fudan.edu.cn

Abstract. The competitive pulse coupled neural network (CPCNN) model is proposed based on the pulse coupled neural network (PCNN), the properties of pulse wave propagation of the CPCNN are analyzed for the solution of network shortest routing. The setting terms of the neuron parameters for the pulse wave propagation along the routing of the shortest path tree (SPT) of network routing graph are educed. The simulation results show that the proposed method is better than Dijkstra algorithm when the network routing graph holds many nodes. With the nodes number of the network increasing, the number of iterations of our algorithm will basically hold the line. The method shows better computational performance and dominance.

Keywords: Competitive PCNN; shortest path tree; Dijkstra algorithm.

1 Introduction

Topology graph of routing is a weighted graph, the weight of routing connected pathway denotes the synthesis cost of information package transmission between two nodes, if the information package is transmitted to destination at the fastest speed, then the shortest synthesis cost path between source node and destination node must be known. The transfer seeking location of information package is usually directly obtained from route table of Router, and the foundation of route table of Router is usually a process of the solution of shortest path tree (SPT), which classical algorithm is Dijkstra algorithm.

Pulse coupled neural network (PCNN), based on the phenomena of synchronous pulse bursts in the animal visual cortex [1], has successfully application in many fields, such as image processing [2], pattern recognition [3], path optimization [4-7]. PCNN can resolve the path optimization problem by using its autowave, and can obtain global optimization result using minimum cost [7], and it is better solution than genetic algorithm or Hopfield network. In 1999, Caulfield and Kinser presented the idea of utilizing the autowave in PCNN to find the solution of the maze problem [7], their method can be used to solve the shortest path of the network, but required neurons were fairly many, each unit path of the graph need corresponding neuron. Afterward, some researchers put forward to resolve the shortest path of the network using DPCNN model and its modified PCNN model [4-6], the number of neurons only correlate with the number of

network nodes, but there are the following some problems to need to be resolved: In [4-7], there were some artificial interference in the iterations process of neuron, all parameters of models are artificially set, and the threshold of neuron was set for one special value, the neuron does not strictly run according to its dynamics equation, so its applied currency will get restriction. In this paper, we put forward a competitive PCNN model based on Johnson PCNN model [8], and the parameters setup of our model have strict theory guidance, and it can automatically obtain the shortest path tree (SPT) of the network routing according to its dynamics equation, the simulation result shows that the proposed method is better than Dijkstra algorithm when the network routing graph holds many nodes. With the nodes number of the network increasing, the number of iteration of our algorithm will basically hold the line. Our algorithm shows better computational performance and dominances.

2 CPCNN Model

We do some modifications based on PCNN model, and called CPCNN model, the CPCNN model is shown in Fig. 1, the equations from (1) to (8) describe this model.

$$S(n) = \min(W_{i_j}, \dots, W_{k_j}), (Y_1 > 0, \dots, Y_k > 0). \quad (1)$$

$$V_j^L(n) = \begin{cases} S(n), & S(n) > 0 \text{ and } S(n) \ll \theta_j^L(n-1) \\ \theta_j^L(n-1), & S(n) > 0 \text{ and } S(n) > \theta_j^L(n-1) \\ V_j^L(n), & S(n) = 0 \end{cases}. \quad (2)$$

$$\theta_j^L(n) = \begin{cases} V_j^L(n) - \Delta_L, & S(n) > 0 \text{ and } V_j^L(n) \geq \Delta_L \\ \theta_j^L(n-1) - \Delta_L, & S(n) > 0 \text{ and } \theta_j^L(n-1) \geq \Delta_L \\ 0, & \text{otherwise} \end{cases}. \quad (3)$$

$$L_j(n) = \theta_j^L(n). \quad (4)$$

$$F_j(n) = I. \quad (5)$$

$$U_j(n) = F_j(n)(1 + \beta_j L_j(n)) = I(1 + \beta_j L_j(n)). \quad (6)$$

$$\theta_j^T(n) = \begin{cases} V_j^T, & Y_j(n) = 1 \\ \theta_j^T(n-1) - \Delta_T, & Y_j(n) = 0 \text{ and } \theta_j^T(n-1) \geq \Delta_T \\ 0, & \text{otherwise} \end{cases}. \quad (7)$$

$$Y_j(n) = \text{step}(U_j(n) - \theta_j^T(n)) = \begin{cases} 1, & U_j(n) > \theta_j^T(n) \\ 0, & \text{otherwise} \end{cases}. \quad (8)$$

The model consists of three parts:

The receptive field consists of L channel and F channel. $S(n)$ is output of least positive value selector in L channel, and is a selectivity output (positive value is valid) of external neurons pulse input. $L_j(n)$ is linearity decay on $[V_j^L(n), 0]$ according to (3), $V_j^L(n)$ is various range coefficient in L channel, Δ_L is decay step and is controlled by $S(n)$. F is the output of I which comes from external sources.

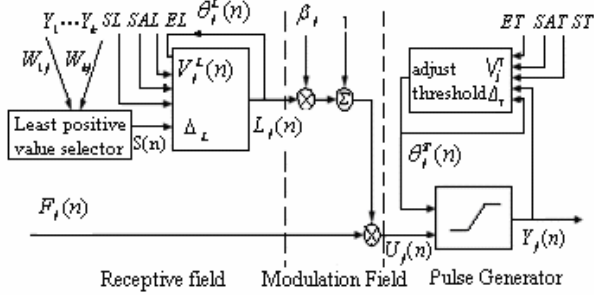


Fig. 1. CPCNN Model

The modulation field, the linking input $L_j(n)$ is added a constant positive bias, then it is multiplied by the feeding input, the bias is unitary, β_j is the linking strength, the total internal activity $U_j(n)$ is the result of modulation.

Pulse generator consists of a threshold adjuster, a comparison organ and a pulse generator. Its mostly function is to generate the pulse output $Y_j(n)$, and adjust threshold value $\theta_j^T(n)$. $\theta_j^T(n)$ is linearity decay on $[V_j^T, 0]$, V_j^T is threshold range coefficient, Δ_T is threshold decay step.

The signal SL and ST are special presetting signals of $\theta_j^L(n)$ and $\theta_j^T(n)$, respectively, SAL and SAT are special presetting signals of range coefficient $V_j^L(n)$ and V_j^T , respectively, EL and ET are presetting energy signals which positive value is valid. Under the energy signals are positive value, we can set initial values or certain special values for $\theta_j^L(n)$ and $\theta_j^T(n)$, $V_j^L(n)$ and V_j^T . The presetting function of CPCNN enhances the controllability of its pulse wave emission.

Compared with Johnson PCNN model, our model changes pulse input sum of external neurons into least positive value selector, this incarnates competitive connection as pulse input, so it is called pulse competition, the selection output $S(n)$ makes less value competition with $\theta_j^L(n-1)$, this is called channel competition, but it holds L channel decay characteristic of PCNN model.

3 Autowave Characteristics of CPCNN

Suppose that sampling time interval of CPCNN model is ΔT , the firing iterations time of the first neuron is n_s , the linking weight between neuron i and neuron j is W_{ij} , the

maximal linking weight is W_{\max} . Define two autowave transmission characteristics, (i) The firing period of neuron j is T_j , (ii) transmission time of autowave is T , which is pass time from the autowave of the first neuron firing emission to spring residual $N-1$ neurons.

Theorem 1. For CPCNN, if each neuron j satisfies

$$\beta_j < 0, V_j^T > I.$$

$$\text{Then the firing period } T_j > \frac{V_j^T - I + \Delta_T}{\Delta_T - I\beta_j\Delta_L} \Delta T.$$

Proof. Suppose the neuron j fires while the k th iteration, after n_0 iterations, it fires again, we have:

$$Y_j(k) = 1, Y_j(k + n_0) = 1.$$

$$\text{So } U_j(k) > \theta_j^T(k) \geq 0, \theta_j^T(k+1) = V_j^T,$$

and that $U_j(k) = I(1 + \beta_j L_j(k))$, according to $\beta_j < 0$, then $0 < U_j(k) \leq I$.

In n_0 iterations, the value of L_j is decay or holds 0, we have:

$$0 < U_j(k + n_0) \leq I.$$

Suppose the value of L_j has n_1 decay in n_0 iterations process, then:

$$U_j(k + n_0) > \theta_j^T(k + n_0) \geq 0.$$

And that $V_j^T > I$, θ_j^T all through decay, according to (6) and (7), we have:

$$I[1 + \beta_j(L_j(k) - n_1\Delta_L)] > V_j^T - (n_0 - 1)\Delta_T,$$

$$\text{namely: } I[1 + \beta_j(L_j(k))] = U_j(k) > V_j^T - (n_0 - 1)\Delta_T + I\beta_j n_1 \Delta_L.$$

$$\text{So that } V_j^T - (n_0 - 1)\Delta_T + I\beta_j n_1 \Delta_L < I, (n_0 - 1)\Delta_T > V_j^T - I + I\beta_j n_1 \Delta_L.$$

And that $0 \leq n_1 \leq n_0$, $n_0 \geq 1$, and $\beta_j < 0$, we have:

$$(n_0 - 1)\Delta_T > V_j^T - I + I\beta_j n_0 \Delta_L.$$

$$\text{So } n_0(\Delta_T - I\beta_j\Delta_L) > V_j^T - I + \Delta_T, T_j = n_0\Delta T.$$

$$\text{So that } T_j > \frac{V_j^T - I + \Delta_T}{\Delta_T - I\beta_j\Delta_L} \Delta T. \text{ The proof is completed.}$$

Theorem 2. For CPCNN, arbitrary neuron j satisfies

$$\beta_j < 0, n_s \leq \frac{W_{\max}}{\Delta_L}, \theta_j^T(0) < I + n_s \Delta_T.$$

Then, the pass iteration time of neuron j from start to accept input pulse to it firing is $n_{0j} \leq \frac{W_{\max}}{\Delta_L}$, the autowave transmission time is described by

$$T \leq \frac{(N-1)W_{\max}}{\Delta_L} \Delta T .$$

Proof. Suppose neuron j receives pulse input from the k th iteration to fire and comes through n_{0j} iterations. According to (1)-(3), after neuron j receives the pulse, the value of L channel starts to decay from $\min(W_{xj}, \theta_1^L(k-1))$, and at best decay to 0, suppose to decay n_1 times. We have

$$n_1 = \frac{\min(W_{xj}, \theta_1^L(k))}{\Delta_L} \leq \frac{W_{\max}}{\Delta_L}, \quad (9)$$

in which x is a neuron which inputs pulse to neuron j .

$$U_j(k+n) = I[1 + \beta_j(\min(\theta_j^L(k-1), W_{xj}) - n_1 \Delta_L)].$$

According to (3) and (8), and $\beta_j < 0$, we have

$$0 < U_j(k+n) \leq I . \quad (10)$$

Because of $\theta_j^T(0) - n_s \Delta_T < I$, and that $\theta_j^T(0)$ at least decays n_s times, if the attenuation of L channel has been completed after n_s times or before n_s times, that $U_j(k+n)=I$, here have

$$n_{0j} = \frac{\min(W_{xj}, \theta_j^L(k))}{\Delta_L} + (n_s - \frac{\min(W_{xj}, \theta_j^L(k))}{\Delta_L}) = n_s, \quad (11)$$

Otherwise, if the attenuation of L channel has been completed at n_s times or after n_s times, we have

$$n_{0j} = n_1 \leq \frac{W_{\max}}{\Delta_L}. \quad (12)$$

According to $n_s \leq \frac{W_{\max}}{\Delta_L}$ and (11)-(12), we have

$$n_{0j} \leq \frac{W_{\max}}{\Delta_L}. \quad (13)$$

Autowave transmission time $T = \Delta T \sum_{k=2}^N n_{0k}$. So that $T \leq \frac{(N-1)W_{\max}}{\Delta_L} \Delta T$. The proof is completed.

Theorem 3. For CPCNN, except starting firing neuron, if arbitrary neuron j satisfies

$$\beta_j < 0, n_s \leq \frac{W_{\max}}{\Delta_L}, \theta_j^T(0) < I + n_s \Delta_T, \theta_j^L(0) > (N-1)W_{\max} + n_s \Delta_L,$$

$$V_j^T > \max\left(\frac{(N-1)(\Delta_T - I\beta_j\Delta_L)W_{\max}}{\Delta_L} + I - \Delta_T, I\right),$$

and the first emitting pulse neuron satisfies the β_j and V_j^T restriction conditions, then autowave transmission path which the first firing neuron fires and brings constitutes the shortest path tree (SPT) of the network.

Proof. Because of $\theta_j^L(0) > (N-1)W_{\max} + n_s\Delta_L$, the finally one firing neuron N has

$$\theta_N^L(0) > W_{xN} + (N-2)W_{\max} + n_s\Delta_L,$$

where x is a neuron which inputs pulse to neuron N , namely

$$\theta_N^L(0) - \left(\frac{(N-2)W_{\max}}{\Delta_L} + n_s\right)\Delta_L > W_{xN}, \theta_N^L(0) > (N-1)W_{\max} + n_s\Delta_L, \quad (14)$$

In CPCNN, the iteration times of the $(N-1)$ th neuron firing is described by

$$M = n_s + \sum_{k=2}^{N-1} n_{0k}.$$

Because of $\beta_j < 0$, $n_s \leq \frac{W_{\max}}{\Delta_L}$, $\theta_j^T(0) < I + n_s\Delta_T$, according to theorem 2, we have

$$M \leq \frac{(N-2)W_{\max}}{\Delta_L} + n_s, \quad (15)$$

$$T \leq \frac{(N-1)W_{\max}}{\Delta_L} \Delta T, \quad (16)$$

According to (14) and (15), we have

$$\theta_N^L(0) - M\Delta_L > W_{xN},$$

namely, $\theta_N^L(M) > W_{xN}$.

Therefore, the first pulse emitting of the finally one firing neuron is evocable by its external neuron pulse input. As a result of the parameter conditions aim at other neurons except the first firing neuron. Therefore, except the first firing neuron, the first firing of other neurons all are aroused by external neurons input pulse.

$$V_j^T > \max\left(\frac{(N-1)(\Delta_T - I\beta_j\Delta_L)W_{\max}}{\Delta_L} + I - \Delta_T, I\right),$$

$$V_j^T > I, \quad (17)$$

$$V_j^T > \frac{(N-1)(\Delta_T - I\beta_j\Delta_L)W_{\max}}{\Delta_L} + I - \Delta_T \quad (18)$$

(18) can be written as the following

$$\frac{V_j^T - I + \Delta_r}{(\Delta_r - I\beta_j\Delta_L)} \Delta T > \frac{(N-1)W_{\max}}{\Delta_L} \Delta T \quad (19)$$

According to $\beta_j < 0$, (16) and theorem 1, we have

$$T_j > \frac{(V_j^T - I + \Delta_r)}{\Delta_r - I\beta_j\Delta_L} \Delta T \quad (20)$$

According to (16), (19) and (20), we get

$$T_j > T.$$

Therefore, within N neurons all fire, each neuron only emit once pulse (except the first neuron fires, the first firing of other neurons all are aroused by external neurons input pulse), and its pulse transmits according to the shortest linking path. Without question, the path of pulse pass will buildup a tree based on the root with the first firing neuron, and the path to each node is all the shortest, namely, a shorest path tree (SPT) is obtained. The proof is completed.

4 Experiment Smulation and Result Analysis

SPT operation was done for Fig. 2, except firing neuron, the parameters setup of other neuron satisfy theorem 3, and it is shown in Table 1. Suppose the start firing neuron tab is N_s . We can make out from the parameters setup of neuron, at the start, the first neuron fire, namely $n_s=0$, the shortest path tree of each node is shown in Table 2.

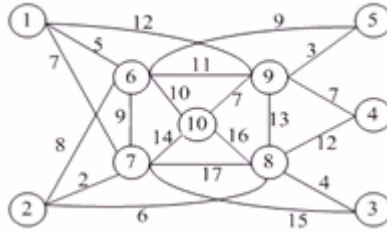


Fig. 2. Route graph of a network

Table 1. Neuron parameters

Neuron	$V_j^L(0)$	$\theta_j^L(0)$	V_j^T	$\theta_j^T(0)$	β_j	I	Δ_L	Δ_r
N_s	156	0	231.5	0	-1	1	1	1
N_j	156	156	231.5	1.9	-1	1	1	1

Table 2. The SPT computation results for the graph in Fig. 2

Source node	SPT	The number of iteration	Convergence
1	1→6→5; 1→6→10; 1→7→2 →8→3; 1→9→4	17	Global
2	2→6→5→9; 2→7→1; 2→8→3; 2→8→4	17	Global
3	3→8→2→6; 3→8→2→7→1; 3→8→4; 3→8→9→5; 3→8→10	18	Global
4	4→8→2→7; 4→8→3; 4→9→1; 4→9→5; 4→9→6; 4→9→10	20	Global
5	5→6→1; 5→6→2; 5→6→7; 5→9→4; 5→9→8→3; 5→9→10	20	Global
6	6→1; 6→2→8→3; 6→5; 6→7; 6→9→4; 6→10	18	Global
7	7→1→9; 7→2→8→3; 7→2 →8→4; 7→6; 7→10	20	Global
8	8→2→6; 8→2→7→1; 8→3; 8→4; 8→9→5; 8→10	16	Global
9	9→1→7; 9→4; 9→5; 9→6→2; 9→8→3; 9→10	27	Global
10	10→6→1; 10→7→2; 10→7→3; 10→9→4; 10→9→5	20	Global

We apply our algorithm to finding the shortest path tree from arbitrarily specified start node to all other nodes in randomly generated undirected and symmetric graphs. The graphs are generated in such a way that the N nodes are uniformly distributed in a square with the edge length $d=50$, and each two nodes is connected with an edge if their distance is no more than $0.3d$. Fig. 3 shows a generated graph with $d=50$ and $N=100$, Fig. 4 shows a SPT graph which was obtained by CPCNN. Compared with Dijkstra algorithm under the graph is uniformly distributed in a square with the edge length $d=150$ and N nodes, and each two nodes is connected with an edge if their distance is no more than $0.5d$, and N chooses different value and the running network times is 300, the results are shown in Table 3.

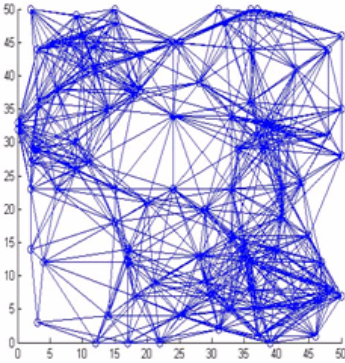


Fig. 3. Random creating network graph

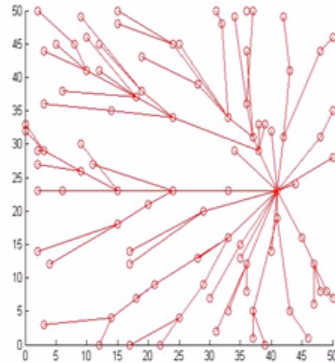


Fig. 4. SPT graph

We can make out from Table 3 that CPCNN algorithm has more obvious characteristics than Dijkstra algorithm: (i) As Δ_L is definite, average number of iteration K of CPCNN algorithm relates with average beeline distance of network, $K = (\sqrt{2d^2 + d/2}) / (2\Delta_L)$, and that the number of iteration of Dijkstra algorithm relates with the number of nodes, with the number of nodes increasing, the number of iteration necessarily increase. (ii) In CPCNN algorithm, the influence of iteration for increasing node of network is very little. (iii) With the number of nodes increasing, the number of iteration for Dijkstra algorithm is more and more increasing. Therefore, CPCNN algorithm shows better computational performance and dominance.

Table 3. The number of iteration comparison of CPCNN and Dijkstra

nodes	Running times	Dijkstra algorithm	CPCNN algorithm
50	300	48	142.3
100	300	98	145.3
150	300	148	148.1
200	300	198	148
300	300	298	147.8
500	300	498	146.3
700	300	698	145
1000	300	998	142.3

5 Conclusions

In this paper, we present a new arithmetic based on CPCNN to solve the shortest path tree problem. Our works are based on the symmetric weighted graph. This is a non-deterministic method which would guarantee the globally solutions. The highly parallel computation of the network will result in fast finding of the shortest paths if the network is realized with VLSI. It is easy to extend the proposed CPCNN to other case when the topology has some slight changes. The simulation results show that the proposed method is better than Dijkstra algorithm when the network routing graph holds many nodes. With increasing the nodes of the network, the number of iteration of our algorithm will basically hold the line. The CPCNN algorithm shows better computational performance and dominance and has important significance in both theory and applications.

Acknowledgments. Supported by Natural Science Fund of Yunnan (2005F0010M, 2007F174M), Youth Fund of Yunnan University (2007Qo24C), China.

References

1. Eckhorn, R., Reitboeck, H.J., Arndt, M., Dicke, P.: Feature Linking via Synchronization among Distributed Assemblies: Simulation of Result from Cat Visual Cortex. *Neural Comput.* 2(3), 293–307 (1990)

2. Gu, X.D., Guom, S.D., et al.: A New Approach for Automated Image Segmentation Based on Unit Linking PCNN. In: Proceedings of 2002 International Conference on Machine Learning and Cybernetics, pp. 175–178. IEEE Press, Beijing (2002)
3. Muresan, R.C.: Pattern Recognition Using Pulse-coupled Neural Networks and Discrete Fourier Transforms. *Neurocomputing* 51, 487–493 (2003)
4. Gu, X.D., Yu, D.H., Zhang, L.M.: Finding the Shortest Path Based on Delay. *Acta Electronica Sinica* 32(9), 1441–1443 (2004) (in Chinese)
5. Song, Y.M., Yuan, D.L.: An Algorithm for Finding the Shortest Path of Labyrinth Based on PCNN Model. *Journal of Circuits and Systems* 10(3), 72–75 (2005) (in Chinese)
6. Zhang, J.Y., Wang, D.F., Shi, M.H.: Finding Shortest Path in Shortest Time with Output Threshold Coupled Neural Networks. *Science in China (Series E)* 33(6), 522–530 (2003)
7. Caufield, H.J., Kinsler, J.M.: Finding Shortest Path in the Shortest Time Using PCNN's. *IEEE Trans. on Neural Networks* 10(3), 604–606 (1999)
8. Johnson, J.L., Padgett, M.L.: PCNN Models and Application. *IEEE Trans. on Neural Networks* 10(3), 480–498 (1999)

Criteria for Exponential Stability of Cohen-Grossberg Neural Networks with Multiple Time-Varying Delays

Anhua Wan¹ and Weihua Mao^{2,3,*}

¹ School of Mathematics and Computational Science,
Sun Yat-sen University, 510275 Guangzhou, China
wananhua@mail.sysu.edu.cn

² Department of Applied Mathematics, College of Science,
South China Agricultural University, 510642 Guangzhou, China

³ College of Automation Science and Engineering,
South China University of Technology, 510641 Guangzhou, China
mao.weihua@mail.scut.edu.cn

Abstract. The exponential stability is analyzed for Cohen-Grossberg neural networks with multiple time-varying delays. The boundedness, differentiability or monotonicity condition is not assumed on the activation functions. Lyapunov functional method is employed to investigate the stability of the neural networks, and general sufficient conditions for the global exponential stability are derived. A numerical example is presented to demonstrate the effectiveness of the obtained criteria.

Keywords: Cohen-Grossberg neural networks, multiple time-varying delays, exponential stability.

1 Introduction

Cohen-Grossberg neural networks model, as an important recurrently connected neural networks model [2], includes many significant models from neurobiology, population biology and evolutionary theory ([6]), for example, Hopfield-type neural networks model ([12]) and Volterra-Lotka biological population model. Meanwhile, the neural networks model has been applied in many important areas such as signal processing, image processing and pattern classification ([6]). Therefore, the study on the neural networks has been the focus of interest ([1, 9, 14-19]).

Due to the finite switching speed of neurons and amplifiers, time delays inevitably exist in biological and artificial neural networks ([1, 11, 16]). In this paper, we consider Cohen-Grossberg neural networks model with multiple time-varying delays described by the following functional differential equations

* Corresponding author. The authors acknowledge the support of Young Teachers Research Support Foundation of Sun Yat-sen University under Grant No. 2007-34000-3171915.

$$\frac{du_i(t)}{dt} = -a_i(u_i(t)) \left[b_i(u_i(t)) - \sum_{k=0}^K \sum_{j=1}^n w_{ij}^{(k)} f_j^{(k)}(u_j(t - \tau_{ij}^{(k)}(t))) + J_i \right], \quad (1)$$

$$i = 1, 2, \dots, n,$$

where $n \geq 2$ is the number of neurons in the neural networks, $u_i(t)$ denotes the neuron state vector, a_i denotes an amplification function, b_i denotes a self-signal function, $W^{(k)} = (w_{ij}^{(k)})_{n \times n}$ denotes connection weight matrix, $f_j^{(k)}$ denotes an activation function, $\tau_{ij}^{(0)}(t) \equiv 0$, $\tau_{ij}^{(k)}(t) \geq 0$ ($k = 1, 2, \dots, K$) denote time-varying delays caused during the switching and transmission processes, and J_i represents the constant external input.

The initial conditions associated with system (1) are of the following form

$$x_i(s) = \phi_i(s) \in C([t_0 - \tau, t_0], R), \quad s \in [t_0 - \tau, t_0], \quad i = 1, 2, \dots, n, \quad (2)$$

where $\tau = \sup \{ \tau_{ij}^{(k)}(t) : t \geq t_0, 1 \leq i, j \leq n, 1 \leq k \leq K \} \in [0, +\infty)$ and $C([t_0 - \tau, t_0], R)$ denotes the space of all real-valued continuous functions defined on $[t_0 - \tau, t_0]$. Denote $\phi(s) = (\phi_1(s), \phi_2(s), \dots, \phi_n(s))^T$.

When in particular $K = 1$, system (1) reduces to (8)

$$\frac{du_i(t)}{dt} = -a_i(u_i(t)) \left[b_i(u_i(t)) - \sum_{j=1}^n w_{ij} f_j(u_j(t)) - \sum_{j=1}^n w_{ij}^{\tau} f_j^{\tau}(u_j(t - \tau_{ij}(t))) + J_i \right], \quad i = 1, 2, \dots, n, \quad (3)$$

where $W = (w_{ij})_{n \times n}$ and $W^{\tau} = (w_{ij}^{\tau})_{n \times n}$ denote the connection weight matrices, f_j and f_j^{τ} denote the activation functions, and $\tau_{ij}(t) \geq 0$.

The initial conditions associated with system (3) are of the following form

$$u_i(s) = \phi_i(s) \in C([t_0 - \tau, t_0], R), \quad s \in [t_0 - \tau, t_0], \quad i = 1, 2, \dots, n, \quad (4)$$

where $\tau = \sup \{ \tau_{ij}(t) : t \geq t_0, 1 \leq i, j \leq n \} \in [0, +\infty)$.

Moreover, system (1) includes many other well-known models, for example,

(1) Hopfield-type neural networks with time-varying delays (5)

$$C_i \frac{du_i(t)}{dt} = -\frac{u_i}{R_i} + \sum_{j=1}^n w_{ij} f_j(u_j(t - \tau_{ij}(t))) + J_i, \quad i = 1, 2, \dots, n. \quad (5)$$

(2) cellular neural networks with time-varying delays (19)

$$\frac{du_i(t)}{dt} = -b_i u_i(t) + \sum_{j=1}^n w_{ij} f_j(u_j(t)) + \sum_{j=1}^n w_{ij}^{\tau} f_j^{\tau}(u_j(t - \tau_{ij}(t))) + J_i, \quad (6)$$

$$i = 1, 2, \dots, n.$$

(3) BAM neural networks with time-varying delays:

$$\begin{cases} \frac{dx_i(t)}{dt} = -a_i x_i(t) + \sum_{j=1}^m w_{ij} f_j(y_j(t - \tau_{ji}(t))) + I_i, & i = 1, 2, \dots, n; \\ \frac{dy_j(t)}{dt} = -b_j y_j(t) + \sum_{i=1}^n v_{ij} g_i(x_i(t - \sigma_{ij}(t))) + J_j, & j = 1, 2, \dots, m, \end{cases} \quad (7)$$

where w_{ji} and v_{ij} respectively denotes connection weight of the j -th neuron in Y -layer to the i -th neuron in X -layer and that of the i -th neuron in X -layer to the j -th neuron in Y -layer, $g_i(x_i)$ and $f_j(y_j)$ respectively denotes output activation of the i -th neuron in X -layer and that of the j -th neuron in Y -layer, $\tau_{ji}(t) / \sigma_{ij}(t)$ respectively denotes delay in Y/X -layer, I_i and J_j respectively denotes external input to the i -th neuron in X -layer and that to the j -th neuron in Y -layer.

(4) Lotka-Volterra recurrent neural networks with time-varying delays([I18](#))

$$\frac{du_i(t)}{dt} = u_i(t) \left[-u_i(t) + \sum_{j=1}^n w_{ij} u_j(t) + \sum_{j=1}^n w_{ij}^{\tau} u_j(t - \tau_{ij}(t)) + J_i \right], \quad i = 1, 2, \dots, n. \quad (8)$$

The stability of neural networks is of crucial importance for the designing and successful applications of neural networks. This paper aims to present new general sufficient conditions for the global exponential stability of [\(II\)](#).

For the delayed neural networks [\(II\)](#), we only make the following assumptions:

(A₁) Each a_i is continuous, and there exist two constants $\hat{\alpha}_i$ and $\acute{\alpha}_i$ such that $0 < \hat{\alpha}_i \leq a_i(r) \leq \acute{\alpha}_i, \forall r \in R$.

(A₂) Each b_i is continuous, and there exists a constant $\lambda_i > 0$ such that for any $r_1, r_2 \in R, (r_1 - r_2)[b_i(r_1) - b_i(r_2)] \geq \lambda_i(r_1 - r_2)^2$.

(A₃) Each $f_j^{(k)}$ is Lipschitz continuous. Denote $m_j^{(k)}$ the minimal Lipschitz constant of $f_j^{(k)}$, namely, $m_j^{(k)} = \sup_{s_1, s_2 \in R, s_1 \neq s_2} |f_j^{(k)}(s_1) - f_j^{(k)}(s_2)| / |s_1 - s_2|$.

Since the monotonicity or boundedness assumption on activation functions will make the results inapplicable to some engineering problems([\(4\)](#), [\(I0\)](#)), boundedness, monotonicity/differentiability condition on $f_j^{(k)}$ are all abandoned.

2 Existence and Uniqueness of Equilibrium of [\(II\)](#)

Firstly, we recall some preliminary concepts. R^n denotes the n -dimensional real vector space. For any $x = (x_1, x_2, \dots, x_n)^T \in R^n, \|x\|_p = (\sum_{i=1}^n |x_i|^p)^{1/p}, p \in [1, +\infty)$ denotes the l^p -norm of x . For any two operators F and G, FG denotes the composition of operators, that is, $FG(x) = F(G(x)), x \in D(G), G(x) \in D(F)$. Let $\text{sign}(r)$ denote the sign function of $r \in R$. We recall the well-known Young inequality: For $a \geq 0, b \geq 0, p > 1, q > 1(\frac{1}{p} + \frac{1}{q} = 1), ab \leq \frac{1}{p}a^p + \frac{1}{q}b^q$.

Since the analysis of the existence and uniqueness of equilibrium of [\(II\)](#) is similar to that of Cohen-Grossberg neural networks with multiple discrete delays(Theorem 1, [\(I3\)](#)), we can readily obtain the following result.

Theorem 1. *Suppose that (A₁)-(A₃) hold. Then for each set of external input J_i , the delayed neural networks [\(II\)](#) has a unique equilibrium point u^* , if there exist $p \in [1, +\infty)$ and positive numbers $d_i, c_i, r_{ij}^{(k)}$ such that*

$$\frac{1}{p} \sum_{k=0}^K \sum_{j=1}^n \left\{ \frac{d_j}{d_i} \left(r_{ji}^{(k)} \right)^{p-1} m_i^{(k)} |w_{ji}^{(k)}| + (p-1) \frac{d_i c_j}{d_j c_i} \left(r_{ij}^{(k)} \right)^{-1} m_j^{(k)} |w_{ij}^{(k)}| \right\} < \lambda_i, \quad i = 1, 2, \dots, n. \quad (9)$$

3 Global Exponential Stability of (II)

In this section, we will investigate the global exponential stability of (II) based on Lyapunov's direct method and delay differential inequality technique.

Lemma 1. (Halany inequality)([3]) *Let λ and μ be constants with $0 < \mu < \lambda$. Let $x(t)(\geq 0)$ be a continuous function on $[t_0 - \tau, +\infty)$ satisfying*

$$x'(t) \leq -\lambda x(t) + \mu \bar{x}(t), \quad t \geq t_0, \quad (10)$$

where $\bar{x}(t) := \sup_{t-\tau \leq s \leq t} x(s)$. Then

$$x(t) \leq \bar{x}(t_0) e^{-r(t-t_0)}, \quad t \geq t_0, \quad (11)$$

where r is the unique positive solution of the equation $r - \lambda + \mu e^{r\tau} = 0$.

Theorem 2. *Suppose that (A₁)-(A₃) hold and there exist $p \in [1, +\infty)$ and positive numbers $d_i, c_i, r_{ij}^{(k)}$ such that*

$$\begin{aligned} a &:= \min_{1 \leq i \leq n} \alpha_i \left\{ p\lambda_i - (p-1) \sum_{k=0}^K \sum_{j=1}^n m_j^{(k)} \frac{d_i c_j}{d_j c_i} (r_{ij}^{(k)})^{-1} |w_{ij}^{(k)}| \right. \\ &\quad \left. - m_i^{(0)} \sum_{j=1}^n \frac{d_j}{d_i} (r_{ji}^{(0)})^{p-1} |w_{ji}^{(0)}| \right\} \\ &> b := \max_{1 \leq i \leq n} \alpha_i \left\{ m_i^{(k)} \sum_{k=1}^K \sum_{j=1}^n \frac{d_j}{d_i} (r_{ji}^{(k)})^{p-1} |w_{ji}^{(k)}| \right\}. \end{aligned} \quad (12)$$

Then for each set of external input J_i , system (II) has a unique equilibrium u^* , which is globally exponentially stable, and there exists a constant $\sigma > 0$ such that the exponential decay estimate of any solution satisfies

$$\|u(t) - u^*\|_p \leq \sqrt[p]{ce}^{-\frac{\sigma}{p}(t-t_0)} \sup_{s \in [t_0 - \tau, t_0]} \|\phi(s) - u^*\|_p, \quad t \geq t_0, \quad (13)$$

where $c = \max_{1 \leq i \leq n} \left\{ \frac{d_i}{\alpha_i} \left(\frac{d_i}{c_i} \right)^{p-1} \right\} / \min_{1 \leq i \leq n} \left\{ \frac{d_i}{\alpha_i} \left(\frac{d_i}{c_i} \right)^{p-1} \right\}$, and σ is the unique positive solution of the equation

$$\sigma - a + be^{\tau\sigma} = 0. \quad (14)$$

Proof. Condition (II2) implies condition (II) holds, and thus it follows from Theorem I that system (II) has a unique equilibrium $u^* = (u_1^*, u_2^*, \dots, u_n^*)^T$.

Let $x_i(t) = d_i c_i^{(1-p)/p} (u_i(t) - u_i^*)$. Denote $x(t) = (x_1(t), x_2(t), \dots, x_n(t))^T$. Substitution of $u_i(t) = c_i^{(p-1)/p} x_i(t) / d_i + u_i^*$ into (II) yields

$$\begin{aligned} \frac{dx_i(t)}{dt} &= -\frac{d_i}{c_i \frac{p-1}{p}} a_i \left(\frac{c_i \frac{p-1}{p}}{d_i} x_i(t) + u_i^* \right) \left\{ b_i \left(\frac{c_i \frac{p-1}{p}}{d_i} x_i(t) + u_i^* \right) - b_i(u_i^*) \right. \\ &\quad \left. - \sum_{k=0}^K \sum_{j=1}^n w_{ij}^{(k)} \left[f_j^{(k)} \left(\frac{c_j \frac{p-1}{p}}{d_j} x_j(t - \tau_{ij}^{(k)}(t)) + u_j^* \right) - f_j^{(k)}(u_j^*) \right] \right\}, \\ &\quad i = 1, 2, \dots, n. \end{aligned} \quad (15)$$

Let $p_i(x_i(t)) = a_i(\frac{c_i^{p-1}}{d_i}x_i(t) + u_i^*)$, $q_i(x_i(t)) = b_i(\frac{c_i^{p-1}}{d_i}x_i(t) + u_i^*) - b_i(u_i^*)$ and $s_j^{(k)}(x_j(t - \tau_{ij}^{(k)}(t))) = f_j^{(k)}(\frac{c_j^{p-1}}{d_j}x_j(t - \tau_{ij}^{(k)}(t)) + u_j^*) - f_j^{(k)}(u_j^*)$. Then

$$\frac{dx_i(t)}{dt} = -\frac{d_i}{c_i^{\frac{p-1}{p}}}p_i(x_i(t))\left\{q_i(x_i(t)) - \sum_{k=0}^K \sum_{j=1}^n w_{ij}^{(k)}s_j^{(k)}(x_j(t - \tau_{ij}^{(k)}(t)))\right\}, \quad (16)$$

$i = 1, 2, \dots, n.$

Clearly, 0 is the unique equilibrium of (16). We define the Lyapunov functional

$$V(x(t)) = \sum_{i=1}^n \int_0^{x_i(t)} p \frac{|s|^{p-1}}{p_i(s)} \text{sign}(s) ds. \quad (17)$$

Estimating the derivative of V along the solution trajectory $x(t)$ of (16), we have

$$\begin{aligned} & \frac{dV(x(t))}{dt} \\ &= -\sum_{i=1}^n p|x_i(t)|^{p-1} \text{sign}(x_i(t)) \frac{d_i}{c_i^{\frac{p-1}{p}}} \left\{ q_i(x_i(t)) - \sum_{k=0}^K \sum_{j=1}^n w_{ij}^{(k)} s_j^{(k)}(x_j(t - \tau_{ij}^{(k)}(t))) \right\} \\ &\leq -p \sum_{i=1}^n |x_i(t)|^{p-1} d_i c_i^{(1-p)/p} \lambda_i c_i^{(p-1)/p} d_i^{-1} |x_i(t)| \\ &\quad + p \sum_{i=1}^n \sum_{k=0}^K \sum_{j=1}^n |w_{ij}^{(k)}| |x_i(t)|^{p-1} \frac{d_i}{c_i^{\frac{p-1}{p}}} m_j^{(k)} \frac{c_j^{p-1}}{d_j} |x_j(t - \tau_{ij}^{(k)}(t))| \\ &= -p \sum_{i=1}^n \lambda_i |x_i(t)|^p + p \sum_{i=1}^n \sum_{j=1}^n \sum_{k=0}^K \left\{ m_j^{(k)} \left[\left(\frac{d_i}{d_j} (r_{ij}^{(k)}) \right)^{p-1} |w_{ij}^{(k)}| \right]^{\frac{1}{p}} |x_j(t - \tau_{ij}^{(k)}(t))| \right. \\ &\quad \left. \left[\left(\frac{d_i c_j |w_{ij}^{(k)}|}{d_j c_i r_{ij}^{(k)}} \right)^{\frac{1}{p}} |x_i(t)| \right]^{p-1} \right\} \\ &\leq -p \sum_{i=1}^n \lambda_i |x_i(t)|^p + p \sum_{i=1}^n \sum_{j=1}^n \sum_{k=0}^K \left\{ m_j^{(k)} \cdot \frac{1}{p} \left[\frac{d_i}{d_j} (r_{ij}^{(k)}) \right]^{p-1} |w_{ij}^{(k)}| |x_j(t - \tau_{ij}^{(k)}(t))|^p \right. \\ &\quad \left. + (p-1) \frac{d_i c_j}{d_j c_i} (r_{ij}^{(k)})^{-1} |w_{ij}^{(k)}| |x_i(t)|^p \right\} \\ &= -\sum_{i=1}^n \left\{ p\lambda_i - (p-1) \sum_{k=0}^K \sum_{j=1}^n \frac{d_i c_j m_j^{(k)}}{d_j c_i r_{ij}^{(k)}} |w_{ij}^{(k)}| - \sum_{j=1}^n m_i^{(0)} \frac{d_i}{d_i} (r_{ji}^{(0)})^{p-1} |w_{ji}^{(0)}| \right\} |x_i(t)|^p \\ &\quad + \sum_{i=1}^n \sum_{k=1}^K \sum_{j=1}^n m_i^{(k)} \frac{d_i}{d_i} (r_{ji}^{(k)})^{p-1} |w_{ji}^{(k)}| |x_i(t - \tau_{ji}^{(k)}(t))|^p \\ &\leq -aV(x(t)) + b\bar{V}(x(t)). \end{aligned}$$

By Lemma 2, we obtain

$$V(x(t)) \leq \bar{V}(x(t_0)) e^{-\sigma(t-t_0)}. \quad (18)$$

From the form of the Lyapunov functional (17), we get

$$\begin{aligned} & \min_{1 \leq i \leq n} \left\{ \frac{1}{\alpha_i} \left(d_i / c_i^{\frac{p-1}{p}} \right)^p \right\} \|u(t) - u^*\|_p^p \\ &\leq V(x(t)) \leq \bar{V}(x(t_0)) e^{-\sigma(t-t_0)} \\ &\leq \max_{1 \leq i \leq n} \left\{ \frac{1}{\alpha_i} \left(d_i / c_i^{\frac{p-1}{p}} \right)^p \right\} \sup_{s \in [t_0 - \tau, t_0]} \|\phi(s) - u^*\|_p^p e^{-\sigma(t-t_0)}. \end{aligned}$$

Corollary 1. *Suppose that (A₁)-(A₃) hold and there exist a set of positive numbers d_i such that*

$$a := \min_{1 \leq i \leq n} \hat{\alpha}_i \left\{ \lambda_i - m_i^{(0)} \sum_{j=1}^n \frac{d_j}{d_i} |w_{ji}^{(0)}| \right\} > b := \max_{1 \leq i \leq n} \hat{\alpha}_i \left\{ m_i^{(k)} \sum_{k=1}^K \sum_{j=1}^n \frac{d_j}{d_i} |w_{ji}^{(k)}| \right\}. \quad (19)$$

Then for each set of external input J_i , system (11) has a unique equilibrium u^ , which is globally exponentially stable, and there exists a constant $\sigma > 0$ such that the exponential decay estimate of any solution satisfies*

$$\|u(t) - u^*\|_1 \leq ce^{-\sigma(t-t_0)} \sup_{s \in [t_0 - \tau, t_0]} \|\phi(s) - u^*\|_1, \quad t \geq t_0, \quad (20)$$

where $c = \max_{1 \leq i \leq n} \frac{d_i}{\hat{\alpha}_i} / \min_{1 \leq i \leq n} \frac{d_i}{\hat{\alpha}_i}$ and σ is the unique positive solution of (14).

Corollary 2. *Suppose that (A₁)-(A₃) hold and there exist three sets of positive numbers $d_i, c_i, r_{ij}^{(k)}$ such that*

$$\begin{aligned} a := \min_{1 \leq i \leq n} \hat{\alpha}_i \left\{ 2\lambda_i - \sum_{k=0}^K \sum_{j=1}^n m_j^{(k)} \frac{d_i c_j}{d_j c_i} (r_{ij}^{(k)})^{-1} |w_{ij}^{(k)}| - m_i^{(0)} \sum_{j=1}^n \frac{d_j}{d_i} r_{ji}^{(0)} |w_{ji}^{(0)}| \right\} \\ > b := \max_{1 \leq i \leq n} \hat{\alpha}_i \left\{ m_i^{(k)} \sum_{k=1}^K \sum_{j=1}^n \frac{d_j}{d_i} r_{ji}^{(k)} |w_{ji}^{(k)}| \right\}. \end{aligned} \quad (21)$$

Then for each set of external input J_i , system (11) has a unique equilibrium u^ , which is globally exponentially stable, and there exists a constant $\sigma > 0$ such that the exponential decay estimate of any solution satisfies*

$$\|u(t) - u^*\|_2 \leq \sqrt{c} e^{-\frac{\sigma}{2}(t-t_0)} \sup_{s \in [t_0 - \tau, t_0]} \|\phi(s) - u^*\|_2, \quad t \geq t_0, \quad (22)$$

where $c = \max_{1 \leq i \leq n} \frac{d_i^2}{\hat{\alpha}_i c_i} / \min_{1 \leq i \leq n} \frac{d_i^2}{\hat{\alpha}_i c_i}$ and σ is the unique positive solution of (14).

(A₃)': Each f_j, f_j^τ is Lipschitz continuous. For convenience, we denote m_j, m_j^τ the minimal Lipschitz constant of f_j, f_j^τ respectively.

Corollary 3. *Suppose that (A₁), (A₂), (A₃)' hold and there exist $p \in [1, +\infty)$ and positive numbers $d_i, c_i, r_{ij}, \tilde{r}_{ij} > 0$ such that*

$$\begin{aligned} a := \min_{1 \leq i \leq n} \hat{\alpha}_i \left\{ p\lambda_i - (p-1) \sum_{j=1}^n \left[m_j \frac{d_i c_j}{d_j c_i} r_{ij}^{-1} |w_{ij}| + m_j^\tau \frac{d_i c_j}{d_j c_i} \tilde{r}_{ij}^{-1} |w_{ij}^\tau| \right] \right. \\ \left. - m_i \sum_{j=1}^n \frac{d_j}{d_i} r_{ji}^{p-1} |w_{ji}| \right\} > b := \max_{1 \leq i \leq n} \hat{\alpha}_i \left\{ m_i^\tau \sum_{j=1}^n \frac{d_j}{d_i} \tilde{r}_{ji}^{p-1} |w_{ji}^\tau| \right\}. \end{aligned} \quad (23)$$

Then for each set of external input J_i , system (3) has a unique equilibrium u^ , which is globally exponentially stable, and there exists a constant $\sigma > 0$ such that the exponential decay estimate of any solution satisfies*

$$\|u(t) - u^*\|_p \leq \sqrt[p]{c} e^{-\frac{\sigma}{p}(t-t_0)} \sup_{s \in [t_0 - \tau, t_0]} \|\phi(s) - u^*\|_p, \quad t \geq t_0, \quad (24)$$

where $c = \max_{1 \leq i \leq n} \left\{ \frac{d_i}{\alpha_i} \left(\frac{d_i}{c_i} \right)^{p-1} \right\} / \min_{1 \leq i \leq n} \left\{ \frac{d_i}{\alpha_i} \left(\frac{d_i}{c_i} \right)^{p-1} \right\}$ and σ is the unique positive solution of the equation $\sigma - a + be^{\tau\sigma} = 0$.

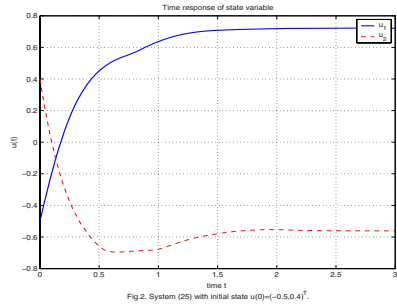
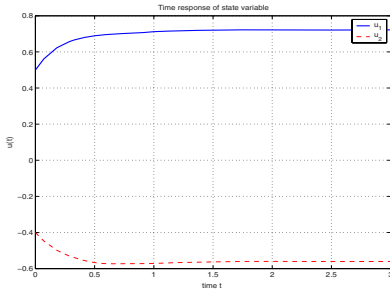
Remark 1. Hwang et al. [7] proved the global exponential stability of (3) with $f_j^\tau = f_j$, but they additionally required that each b_i is differentiable and f_i is bounded. Jiang [8] proved the global exponential stability of (3), but he additionally required that each $\tau_{ij}(t)$ be differentiable and $\sup_{t \in R} \tau_{ij}'(t) = \tau^* < 1$.

4 Numerical Simulation

Example 1. Let $f_1(r) = 0.4r + \cos 0.6r$, $f_2(r) = 0.2r - \sin 0.8r, \forall r \in R$.

$$\begin{cases} \frac{du_1(t)}{dt} = -(2 + \sin u_1(t)) [2u_1(t) - 0.3f_1(u_1(t - |\sin t|)) \\ \quad - 0.264f_2(u_2(t - |\cos t|)) + J_1] \\ \frac{du_2(t)}{dt} = -(3 + \cos u_2(t)) [u_2(t) - 0.3f_1(u_1(t - |\sin 2t|)) \\ \quad - 0.25f_2(u_2(t - |\cos 2t|)) + J_2] \end{cases} \quad (25)$$

It is easy to verify that $(A_1), (A_2), (A_3)'$ are satisfied with $\alpha_1 = 3, \alpha_1 = 1, \alpha_2 = 4, \alpha_2 = 2, \lambda_1 = 2, \lambda_2 = m_1 = m_2 = \tau = 1$. Since f_1, f_2 are unbounded and $\tau_{ij}(t)$ are not differentiable, the criteria in [7], [8] are not satisfied. On the other hand, if we set $p = 1, d_1 = 1, d_2 = 1.1$, then (23) is satisfied, since $a = \min_{1 \leq i \leq 2} \alpha_i \lambda_i = 2 > b = \max_{1 \leq i \leq 2} \alpha_i m_i \sum_{j=1}^2 \frac{d_j}{d_i} |w_{ji}| = 1.96$. Therefore, by Corollary 3, we deduce that for any inputs J_1, J_2 , (25) has a unique equilibrium point u^* , which is globally exponentially stable, and the exponential decay estimate of any solution satisfies (24), where $c = \frac{40}{11}$ and $\sigma = 0.0135$. For numerical simulation, we set $J_1 = -1, J_2 = 1$. Here we consider two cases: In the first case (Fig. 1), $(u_1(0), u_2(0))^T = (0.5, -0.4)^T$; In the second case (Fig. 2), $(u_1(0), u_2(0))^T = (-0.5, 0.4)^T$. The simulation results show that the unique equilibrium point $u^* = (0.7219, -0.5607)^T$ is globally exponentially stable.



5 Conclusions

In this paper we discuss the exponential stability of Cohen-Grossberg neural networks model with multiple time-varying delays. Only assuming the activation

functions to be Lipschitz continuous, we derive new sufficient conditions for the global exponential stability of model (III), which improve some existing results.

References

1. Chen, T.P., Rong, L.B.: Delay-independent Stability Analysis of Cohen-Grossberg Neural Networks. *Physics Letters A* 317, 436–449 (2003)
2. Cohen, M.A., Grossberg, S.: Absolute Stability and Global Pattern Formation and Partial Memory Storage by Competitive Neural Networks. *IEEE Transactions on Systems, Man and Cybernetics SMC-13*, 815–826 (1983)
3. Driver, R.D.: *Ordinary and Delay Differential Equations*. Springer, New York (1977)
4. van den Driessche, P., Zou, X.: Global Attractivity in Delayed Hopfield Neural Network Models. *SIAM J. Appl. Math.* 58, 1878–1890 (1998)
5. Gopalsamy, K., He, X.Z.: Stability in Asymmetric Hopfield Nets with Transmission Delays. *Physica D* 76, 344–358 (1994)
6. Grossberg, S.: Nonlinear Neural Networks: Principles, Mechanisms, and Architectures. *Neural Networks* 1, 17–61 (1988)
7. Hwang, C.C., Cheng, C.J., Liao, T.L.: Globally Exponential Stability of Generalized Cohen-Grossberg Neural Networks with Delays. *Physics Letters A* 319(1-2), 157–166 (2003)
8. Jiang, L.: Global Exponential Stability of Cohen-Grossberg Neural Networks with Time-Varying Delays. *Chaos, Solitons and Fractals* 26, 935–945 (2005)
9. Liao, X.F., Li, C.G., Wong, K.W.: Criteria for Exponential Stability of Cohen-Grossberg Neural Networks. *Neural Networks* 17, 1401–1414 (2004)
10. Morita, M.: Associative Memory with Non-monotone Dynamics. *Neural Networks* 6(1), 115–126 (1993)
11. Peng, J.G., Qiao, H., Xu, Z.B.: A New Approach to Stability of Neural Networks with Time-varying Delays. *Neural Networks* 15, 95–103 (2002)
12. Tank, D.W., Hopfield, J.J.: Simple “Neural” Optimization Networks: An A/D Converter, Signal Decision Circuit, and a Linear Programming Circuit. *IEEE Transactions on Circuits and Systems* 33(5), 533–541 (1986)
13. Wan, A.H., Mao, W.H., Qiao, H., Zhang, B.: Global Asymptotic Stability of Cohen-Grossberg Neural Networks with Multiple Discrete Delays. In: Huang, D.-S., Heutte, L., Loog, M. (eds.) *ICIC 2007*. LNCS (LNAI), vol. 4682, pp. 47–58. Springer, Heidelberg (2007)
14. Wang, L., Zou, X.F.: Exponential Stability of Cohen-Grossberg Neural Networks. *Neural Networks* 15, 415–422 (2002)
15. Wang, L., Zou, X.F.: Harmless Delays in Cohen-Grossberg Neural Network. *Physica D* 170(2), 162–173 (2002)
16. Wu, W., Cui, B.T., Lou, X.: Some Criteria for Asymptotic Stability of Cohen-Grossberg Neural Networks with Time-Varying Delays. *Neurocomputing* 70(4-6), 1085–1088 (2007)
17. Ye, H., Michel, A.N., Wang, K.: Qualitative Analysis of Cohen-Grossberg Neural Networks with Multiple Delays. *Physics Review E* 51, 2611–2618 (1995)
18. Zhang, Y., Tan, K.K.: Dynamic Stability for Lotka-Volterra Recurrent Neural Networks with Delays. *Physical Review E* 66, 011910 (2002)
19. Zhou, D.M., Cao, J.D.: Globally Exponential Stability Conditions for Cellular Neural Networks with Time-Varying Delays. *Applied Mathematics and Computation* 131(2-3), 487–496 (2002)

Detecting Unsuccessful Automated Negotiation Threads When Opponents Employ Hybrid Strategies

Ioannis Papaioannou, Ioanna Roussaki, and Miltiades Anagnostou

National Technical University of Athens, School of Electrical and Computer Engineering,
9 Heroon Polytechniou Str, 157-73 Athens, Greece
{jpapai, nanario, miltos}@telecom.ntua.gr

Abstract. In artificial intelligence systems, building agents that negotiate on behalf of their owners aiming to maximise their utility is a quite challenging research field. In this paper, such agents are enhanced with techniques based on neural networks (NNs) to predict their opponents' hybrid negotiation behaviour, thus achieving more profitable results. The NNs are used to early detect the cases where agreements are not achievable, supporting the decision of the agents to withdraw or not from the negotiation threads. The designed NN-assisted negotiation strategies have been evaluated via extensive experiments and are proven to be very useful.

Keywords: Automated negotiations; MLP & GR neural networks; NN-assisted negotiation strategies; Opponent behaviour prediction; Early detection of unsuccessful negotiations; Hybrid negotiation strategies.

1 Introduction

Automated negotiations are gradually gaining momentum in the artificial intelligence research domain [1]. In this field, building intelligent agents adequate for participating in negotiations and acting autonomously on behalf of their owners is a quite complex and challenging task [2]. Among the research objectives in this field, lies the design of techniques to enforce these agents with capabilities and rationale that resembles the human behaviour [1]. In negotiation procedures, the participants aim to satisfy their human or corporate owners as much as possible, acting in environments of scarce resources. In automated negotiations, this role is assigned to intelligent agents that have conflicting interests and a desire to cooperate in order to reach a mutually acceptable agreement [3]. These agents act based on predefined rules and procedures that are specified by the employed negotiation protocol [4]. The negotiating agents use a reasoning model based on which their response to their opponent's offers are formulated [5]. This reasoning model is called negotiation strategy of the agents [6].

The work presented in this paper is mainly concerned with the design and evaluation of optimised negotiation strategies for autonomous intelligent agents. The proposed strategies are applicable to single-issue bilateral negotiation frameworks, where agents have strict deadlines. The client agents employ learning techniques based on MLP and GR Neural Networks (NNs), aiming to predict their opponents' behaviour

and detect the outcome of each negotiation thread early during the negotiation process. The negotiation strategies employed by the opponent agent are characterized as hybrid in this work, as they are formulated as the weighted sum of the main two kinds of strategies, i.e. the time-dependent and the behavior-dependent ones, originally presented in [7]. The resulting strategy varies from time-dependent to behavior-dependent, while any combination in-between is possible. Similar approaches of mixed strategies were not that popular in the past, but have recently started to attract the attention of the research community. In [8], the authors identify the mixed strategies as an efficient option in e-negotiations, while in [9] the mixed strategies are characterized as more profitable, if properly applied on e-negotiations. Moreover in [10] and [11], an attempt is made to approximate the curve that fits the opponent's offers, which employs a combination of time-dependent and behavior-dependent tactics that may thus be characterized as mixed or hybrid strategies.

The rest of the paper is structured as follows. In Section 2, a formal statement of the problem studied is provided, along with a description of the negotiation strategy space of the Provider Agent. Section 3 presents the NN-assisted negotiation strategy of the Client Agent, while Section 4 elaborates on the architecture and configuration of the NNs employed. Section 5 describes the experiments conducted, summarizes the obtained results and evaluates the proposed approaches. A brief overview of the related work is provided in Section 6. Finally, in Section 7 conclusions are drawn and future research plans are exposed.

2 Negotiation Problem and Strategies

2.1 Negotiation Problem Description

The research presented in this paper studies a single issue, bilateral automated negotiation framework. Thus, there are two negotiating parties (Client and Provider) represented by mobile intelligent agents. The agents negotiate based on an alternating offers protocol [1] aiming to maximize the utilities of the parties they represent.

Let the negotiation process be initiated by the Client Agent (CA) that sends to the Provider Agent (PA) an initial Request for Proposal (RFP) specifying the features of the service/product its owner is interested to obtain. Without loss of generality, it is assumed that the issue under negotiation is the good's price. Thus, the PA negotiates aiming to agree on the maximum possible price, while the CA aims to reduce the price as much as possible. Once the PA receives the RFP of the CA, it either accepts to be engaged in the specific negotiation thread and formulates an initial price offer, or rejects the RFP and terminates the negotiation without a proposal. At each round, the PA sends to the CA a price offer that is subsequently evaluated by the CA against its constraints and reservation values. Then, the CA generates a counter-offer and sends it to the PA that evaluates it and sends another counter-offer to the CA. This process continues until a mutually acceptable offer is proposed by one of the agents, or one of the negotiators withdraws from the negotiation, in case for example its time deadline is reached without an agreement being in place. Thus, at each negotiation round, the agents may: (i) accept the previous offer, in case their constraints are addressed, (ii) generate a counter-offer, or (iii) withdraw from the negotiation.

Quantity p_l^a denotes the price offer proposed by negotiating agent a during negotiation round l . A price proposal p_l^b is always rejected by agent a if $p_l^b \notin [p_m^a, p_M^a]$, where $[p_m^a, p_M^a]$ denotes agent- a 's acceptable price interval. In case an agreement is reached, we call the negotiation successful, while in case one of the negotiating parties quits, it is called unsuccessful. In any other case, the negotiation thread is called active. The objective of the problem studied in this paper is to predict the PA's behaviour in the future negotiation rounds until the CA's deadline expires. More specifically, the negotiation problem studied can formally be stated as follows.

Given: (i) two negotiating parties: a Provider offering a specific good and a Client interested in this good's acquisition, (ii) the acceptable price interval $[p_m^C, p_M^C]$ for the Client, (iii) a deadline T_C up to which the Client must have completed the negotiation with the Provider, (iv) the final negotiation round index L_C for the Client, (v) a round threshold L_C^{dec} until which the Client must decide whether to continue being engaged in the negotiation thread or not, and (vi) the vector $P_l^P = \{p_l^P\}$, where $l = 2k - 1$ and $k = 1, \dots, \left\lfloor \frac{L_C^{dec}}{2} \right\rfloor$, of the prices that were proposed by the Provider during the initial $L_C^{dec} - 1$ negotiation rounds, *find* (i) the price $p_{L_C-1}^P$, that will be proposed by the Provider right before the Client's deadline expires, and (ii) decide on whether the specific negotiation thread can lead to an agreement until round L_C or not.

2.2 Negotiation Strategy Space of the Provider Agent

The policy employed by the negotiating agents in order to generate a new offer is called *negotiation strategy*. There are various such strategies that are described in detail in [7]. These strategies are well defined functions that may use various input parameters in order to produce the value of the issue under negotiation to be proposed at the current negotiation round. Three generic families of autonomous negotiation strategies can be distinguished: time-dependent (TD), resource-dependent and behaviour-dependent (BD) strategies.

In the current version of this study, the PA adopts a linear combination of a TD strategy and a BD strategy, the principles of which are briefly presented hereafter. The time-dependent strategies force agents to concede to their price reservation value as the remaining negotiation time is reduced. The function that provides the price offer p_l^a of agent a at round l and time t_l is:

$$p_l^a = \begin{cases} p_m^a + f^a(t_l)(p_M^a - p_m^a), & \text{if utility is decreasing} \\ p_M^a + [1 - f^a(t_l)](p_M^a - p_m^a), & \text{if utility is increasing} \end{cases}, \text{ where } f^a(t_l) \text{ can be either}$$

polynomial or exponential, i.e., $f^a(t_l) = k^a + (1 - k^a) \left(\frac{\min(t_l, T_a)}{T_a} \right)^{\frac{1}{\beta}}$ or

$f^a(t_l) = e^{\left[1 - \frac{\min(t_l, T_a)}{T_a} \right]^{\beta} \ln k^a}$ respectively. $k^a \in [0, 1]$ determines the initial offer at $t = 0$, T_a is the agent's time deadline, while $\beta > 0$ is the concession rate. Depending on the

specific value of β , three strategy types can be distinguished: *Boulware* ($\beta < 1$) where the agent sticks to its initial offer until the deadline is close to expire [12], *Conceder* ($\beta > 1$) where the agent starts conceding to its reservation value fairly quickly [13], and *Linear* ($\beta = 1$) where the agent concedes by the same amount at each negotiation round. In this paper, both the polynomial and the exponential versions of the time-dependent strategies have been employed as components of the PA's hybrid strategy, for a wide variety of concession rates and expiration deadlines.

The BD strategies generate the subsequent offer based on the opponent's behaviour. Depending on the type of imitation they perform, three categories of such strategies can be distinguished: *relative tit-for-tat* (RTFT), where the agent imitates proportionally the behaviour its opponent demonstrated δ steps ago, *random absolute tit-for-tat* (RATFT), which is the same as RTFT but in absolute terms, and *average tit-for-tat* (ATFT), where the agent imitates proportionally the average behaviour its opponent demonstrated during its γ past offers. The offer generating function for the behaviour dependent strategies is: $p_{l+1}^a = \min(\max(g^a(t_l), p_m^a), p_M^a)$, where

$g^a(t_l) = \frac{p_{l-2\delta}^b}{p_{l-2\delta+2}^b} p_{l-1}^a$, $g^a(t_l) = p_{l-1}^a + p_{l-2\delta}^b - p_{l-2\delta+2}^b + (-1)^s R(M)$ (where s is 0 or 1

if the utility function is decreasing or increasing, and $R(M)$ is a random number

generator in the range $[0, M]$), or $g^a(t_l) = \frac{p_{l-2\gamma}^b}{p_l^b} p_{l-1}^a$ for the RTFT, RATFT, or

ATFT respectively. In this paper, all these three versions of the behaviour-dependent strategies have been employed as components of the PA's hybrid strategy.

Thus, the offer generating function for the hybrid negotiation strategy of the PA is as follows: $p_l^p = w \cdot \{p_m^p + [1 - f^p(t_l)](p_M^p - p_m^p)\} + (1-w) \cdot \min(\max(g(t_{l-1}), p_m^p), p_M^p)$, where w expresses the significance of the time-dependent component in the PA's strategy, while $f^p(t_l)$ can either be expressed by the polynomial or exponential formulation aforementioned and $g^p(t_l)$ is expressed by one of the RTFT, RATFT, or ATFT formulations. Therefore, 6 different couplings exist for the hybrid negotiation strategy of the PA, i.e. polynomial&RTFT, polynomial&RATFT, polynomial&ATFT, exponential&RTFT, exponential&RATFT, and exponential&ATFT.

3 The NN-Assisted Negotiation Strategy of the Client Agent

The mechanism proposed in this paper enhances with learning techniques based on Neural Networks (NNs) any of the legacy strategies [7]. As already stated, the NN-assisted strategies are used by the CA in order to estimate the future behaviour of the PA. The objective is to decide at an early round (i.e. L_C^{dec}) whether to aim for an agreement with the specific PA, or withdraw from the negotiation thread as early as possible, if no agreement is achievable. Thus, the CA is concerned with estimating the PA's price offer upon the expiration of the CA's deadline. For this purpose, Neural Networks (NNs) are employed that are trained off-line with proper training sets and are then used during the on-line negotiation procedure whenever the CA requires so.

The CA negotiates based on a linear time-dependent strategy until round L_C^{dec} , i.e. the CA's price offer function is: $p_l^C = p_m^C + \frac{t_l}{T_C}(p_M^C - p_m^C)$. Then, the CA makes use of the NNs to obtain an estimation $\overline{p_P^{l=L_C}}$ for the PA's price offer upon the expiration of the CA's deadline. Round L_C^{dec} will be hereafter called the *prediction round*. After extensive experimentation, we selected to set the prediction round equal to 50 in the studied framework, as it was rendered as the most efficient round, being at the same time a relatively early one (50/100), while by then, there are enough samples available in order for the NN to predict the PA's behaviour.

Upon round L_C^{dec} the CA decides whether to continue being engaged in the negotiation thread aiming for an agreement before the expiration of its deadline, or withdraw from the negotiation in case an agreement is not feasible given the CA's deadline and acceptable price interval. The criterion used to make the latter decision will simply be whether the negotiation thread is foreseen to be unsuccessful or not. Using the notation introduced in subsection 2.1, and if $D \in \{\text{continue, withdraw}\}$ is the decision taken by the CA at round L_C^{dec} , while $\overline{p_P^{l=L_C}}$ is the estimated by the NN PA's price offer during the CA's final negotiation round, then the decision criterion can be formally expressed as follows: $D = \begin{cases} \text{continue, if } \overline{p_P^{l=L_C}} \leq p_M^C \\ \text{withdraw, if } \overline{p_P^{l=L_C}} > p_M^C \end{cases}$.

Thus, the NN enables the CA to save time and withdraw early from negotiation threads that will not result in agreements. If the CA decides to continue negotiating with this PA, then it uses the legacy negotiation strategy of its choice. The NN is not used by the CA after the decision round L_C^{dec} .

4 The Neural Networks Employed

Neural Networks (NNs) are complex structures that can be trained to approximate the responses originating from most of the physical or not systems. Used in several real world applications, they can be designed to provide reliable output estimations when triggered with any kind of input depending on the problem to be solved. For this reason the training phase is essential. In our framework, where the value prediction for a continuous function is required, we selected to study two types of NNs: the multilayer perceptron (MLP) NN and the Generalized Regression (GR) NN. The latter is more appropriate for on-line function approximation and is a special case of a Radial Basis Function (RBF) NN where an additional linear hidden layer is used [14].

In the studied framework, the NNs are used by usually resource limited autonomous agents, and thus the size of the NNs needs to be reduced. For the same reason, the time required for prediction and the storage resources required by the NNs need to be very low, while the NNs' estimation accuracy needs to be significantly high. Additionally, the instantiation and training phases of networks' creation should be performed offline before the negotiation procedures take place and only once, as these

are the most time consuming. Driven by the design principles above, in the remainder of this section we focus on reasoning over the specific characteristics and configuration of the NNs that have eventually been employed in our experiments.

For the MLP, we used a training function based on the Levenberg-Marquardt algorithm [14] that is the most convenient one for such problems. Each training vector forms the history of PA's offers until the round that is the 50% of the CA's deadline (here $L_C^d = 50$). Thus, the MLP NN can be used at round 50 to provide the prediction for the PA's last round offer. The set of training vectors derives from the application of different values for parameters β , L_p and w on each of the 6 different cases of hybrid strategies described on Section 2.2. The input vectors for the MLP designed, are generated for the following values of the specified combination of strategies to form the output offer each time: $\beta = [0.5 \ 0.8 \ 1 \ 3 \ 5]$, $k^a = [0]$, $L_p = [120:20:220]$ and $W = [0:0.5:1]$. This training space has been selected among many others as it presents the lowest mean square estimation error. Thus, $5 \times 1 \times 6 \times 3 \times 6$ (hybrid combination cases) = 540 different vectors have been applied, each for 100 epochs. The 540 values of the PA's offer on the CA's timeout are the target of the MLP's training. Extensive experiments for the MLP architecture, where the estimation efficiency of various different network configurations has been validated, indicated that the most suitable architecture for this MLP NN is: 27 neurons on the single hidden layer (log-sigmoid transfer function) and 1 output neuron (linear). Similarly to the MLP training, we used for the GR training input vectors of 50 values of the PA's offers (one vector on each of the cases below). Thus, the training vectors are derived from all the possible combinations of the following targets: $\beta = [0.5 \ 0.75 \ 1 \ 1.5 \ 2]$, $k^a = [0]$, $L_p = [150:25:250]$ and $W = [0:0.5:1]$. Note here that the required vectors ($5 \times 1 \times 5 \times 3 \times 6 = 450$) lead to a network with much more neurons (i.e. 450) than the corresponding MLP (1 for each pattern). This was expected, as RBF (and thus GR) NNs tend to have bigger sizes compared to MLP NNs for the same problem. After exhaustive experiments, we selected the spread parameter to be equal to 10 instead of the default (1.0), in order to fit data smoothly and not precisely as similar input vectors can lead to substantially different PA timeout offers and a precise fit would create a network that would suffer from wrong generalization [14]. These two networks having simulated over a wide range of experimental settings for various negotiation characteristics leading to the results presented in the following section.

5 Experimental Evaluation

In this section, the experiments conducted to evaluate the performance of the designed MLP and the GR NNs concerning the estimation of the future behaviour of the negotiating PA are presented. In this respect, various experiment families have been conducted, where the NNs are trained to approximate the behaviour of the PA and attempt to estimate the price offer of the PA at the 100th negotiation round i.e. upon the expiration of the CA's deadline, based on the PA's offers of the first 50 rounds. All families aim to compare the actual behaviour of the PA with the one predicted by the MLP and the GR NNs, when the hybrid PA's strategy is governed by the

following parameter intervals: $[p_m^p, p_M^p] = [100, 200]$, $\beta \in [0.5:0.05:0.9, 1:0.1:2.5]$, $L_p = [150:10:250]$ and $w = [0:0.2:1]$, while the time-dependent CA's strategy has been configured so that: $[p_m^c, p_M^c] = [50, 150]$, $\beta = 1$ and $L_c = 100$. These values were carefully selected in order to feed the NN processes with enough data and efficiently simulate the behaviour of the negotiators when they face no strict agreement deadlines. Additionally, the acceptable price intervals for each party were selected so that a substantial (50%) but not full price interval overlap between the negotiators exists and to constitute the behavioural strategies as fair as possible. As already stated, the time-dependent part of the PA's strategy is formulated either based on the polynomial or the exponential expression presented in Section 2.2, while the behavioural part of its strategy is either the relative tit-for-tat (RTFT), the random absolute tit-for-tat (RATFT), or the average tit-for-tat (ATFT), also described in Section 2.2. Thus, in total, approximately 10 thousand experiments have been conducted with different PA strategy settings. The estimated by the NNs parameter is mainly the minimum PA price offer until the expiration of the CA's deadline, i.e. the PA's offer at the 100th negotiation round, where the procedure should end as the deadline of the CA is reached.

As already stated, the enhanced strategies use the NN estimation for the minimum acceptable price of the PA to decide whether they should continue being engaged in the specific negotiation thread or not. In case $\overline{p_m^p} > p_M^c$, where p_m^p is the price offer made by the PA to the CA upon the CA's deadline expiration (in our study at round 100), the CA terminates the negotiation at round 50. However, after several experiments it has been made clear that as the two NNs have the tendency to either over- or under-estimate the minimum PA price offer, the criterion $\overline{p_m^p} > p_M^c = 150$ is not the one producing the optimal results. In Figure 1, the results of the experiments performed to identify the optimal classification threshold for the success or failure of the negotiation threads are illustrated. More specifically, the respective diagrams depict three lines over varying decision price thresholds: the blue line represents the percentage of the successful negotiation threads (SNTs) that have been mistakenly identified as unsuccessful by the NN, the red line depicts the percentage of the unsuccessful negotiation threads (UNTs) that have been wrongly detected as successful, and the green line illustrates the combined/average error for both cases above. Figure 1 presents the results in case the MLP NN is used, while Figure 2 corresponds to the case where the GR NN is used.

As indicated in Figure 1, even though the 150 price threshold results in very low overall negotiation thread classification error (NTCE) in case the MLP NN is used by the CA, the actual minimum NTCE is observed when the decision criterion for identifying the UNTs is $\overline{p_{L=100}^p} > 146$. This is due to the fact that the MLP NN has a tendency to underestimate the PA price offers. On the other hand, as can be observed in Figure 2, when the GR NN is used by the CA, the minimum NTCE is observed when the decision criterion for classifying a negotiation thread as unsuccessful is $\overline{p_{L=100}^p} > 153$. This is due to the fact that the GR NN often overestimates the PA's minimum price offer.

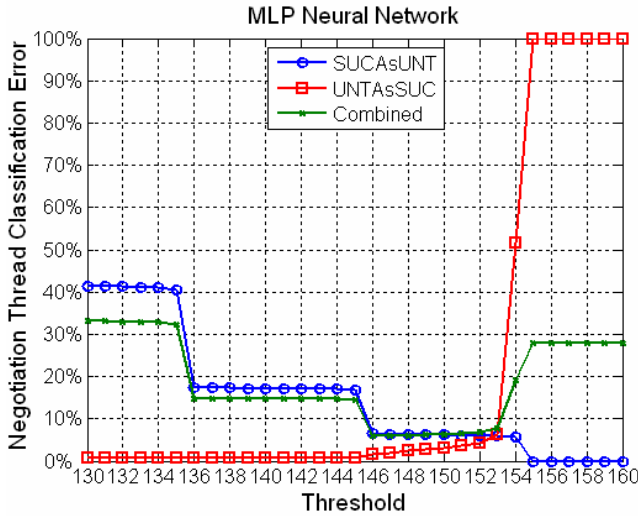


Fig. 1. Negotiation thread classification error over varying decision price threshold in case the CA employs an MLP NN

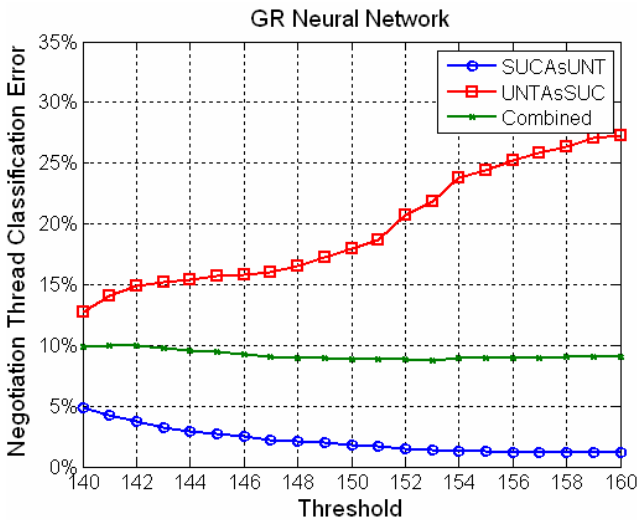


Fig. 2. Negotiation thread classification error over varying decision price threshold in case the CA employs a GR NN

In Table 1, comparative results concerning the unsuccessful negotiation thread (UNT) detection by the two NN-assisted negotiation strategies, when the PA adopts an hybrid negotiation strategy composed by a time-dependent and a behaviour-dependent part. Each table row represents one experiment set where the PA adopts a specific strategy coupling, which is one of the following: Poly&RTFT,

Poly&RATFT, Poly&ATFT, Exp&RTFT, Exp&RATFT, and Exp&ATFT. The values presented are average over the various PA concession rates ($\beta \in [0.5:0.05:0.9, 1:0.1:2.5]$), timeouts ($L_p = [150:10:250]$), and PA strategy weights ($w = [0:0.2:1]$). As already stated, for all experiments it stands that $[p_m^p, p_M^p] = [100, 200]$, $[p_m^c, p_M^c] = [50, 150]$ and $L_c = 100$, while the CA adopts a linear time-dependent strategy until the 50th round, where it decides whether to withdraw or not from the negotiation. The second column in Table 1 depicts the number of unsuccessful negotiation threads (UNTs). These unsuccessful negotiations are due to the fact that $p_{L=100}^p > p_M^c$. The third column in Table 1 indicates that the duration of the UNTs is always equal to $L_c = 100$ ¹ in case no opponent behaviour prediction mechanism is used. The next pair of columns illustrates the number of UNTs that were detected by the NNs at round 50, while the subsequent pair of columns presents the UNTs' elimination ratio, i.e. the ratio of UNTs that were correctly identified by the NNs as unsuccessful and terminated before the expiration of the CAs deadline. The last two pairs of columns in Table 1 illustrate the mean duration of the UNTs and the mean UNT duration decrease with regards to the case where no opponent behaviour prediction mechanism is used.

Table 1. Comparative results concerning the unsuccessful negotiation thread (UNT) detection by the two NN-assisted negotiation strategies, when the PA adopts a hybrid negotiation strategy. Each table row represents one experiment set where the PA adopts a specific strategy coupling indicated in the first table column.

PA Hybrid Strategy Coupling	#UNTs	Mean UNT duration (no NN)	#UNTs detected at round 50		UNTs' elimination ratio (%)		Mean UNT duration		Mean UNT duration decrease (%)	
			MLP	GR	MLP	GR	MLP	GR	MLP	GR
Poly&RTFT	145	100	144	123	99.3	84.8	50.3	57.6	49.7	42.4
Poly&RATFT	145	100	145	132	100	91.0	50	54.5	50	45.5
Poly&ATFT	145	100	145	123	100	84.8	50	57.6	50	42.4
Exp&RTFT	777	100	777	540	100	69.5	50	65.3	50	34.7
Exp&RATFT	777	100	702	540	90.3	69.5	54.8	65.3	45.2	34.7
Exp&ATFT	777	100	777	540	100	69.5	50	65.3	50	34.7
OVERALL					97.3	72.2	51.4	63.9	48.6	36.1

As one may easily observe in Table 1, the MLP NN manages to detect and eliminate 97.3% of the UNTs, demonstrating mean UNT duration equal to 51.4, which is 48.6% lower than the one achieved when no NN is used. On the other hand, the GR NN is less successful, as it detects and eliminates only 72.2% of the UNTs in average, which results in mean UNT duration equal to 63.9, thus reducing it by 36.1% compared to the case where NNs are not used. The reduction of the mean UNT duration achieved by both NNs is highly significant as the CA has the time to get engaged in another negotiation thread that may lead to agreements. Therefore, with regards to the

¹ To be more accurate, the duration of UNTs is equal to: $\min(L_c, L_p)$. However, in this paper's study, we always have $L_c < L_p$, and thus the duration of UNTs is equal to L_c .

elimination of the UNTs, the MLP-assisted strategy clearly outperforms the GR-assisted negotiation strategy, as it detects approximately 35% more UNTs.

The results presented in Table 1 concerning the detection of the UNTs are not enough to evaluate the overall performance of the NN-assisted strategies. This is due to the fact that, even if the MLP NN manages to eliminate up to 35% more UNTs than the GR NN, in several cases it mistakes threads that can lead to agreements for UNTs. In Table 2, evaluation metrics concerning the correct or faulty detection of both SNTs and UNTs by the two NNs are presented. As in Table 1, also in Table 2, each row represents one experiment set where the PA adopts a specific strategy coupling. All values depicted in the rest of the columns of Table 2 are average over the various PA concession rates, timeouts, and PA strategy weights, while the same agent's price intervals, CA timeout and CA decision round have been used as in Table 1. The second and third columns in Table 2 depict the overall number of SNTs and UNTs respectively that have been observed for the strategy coupling indicated in column 1. Of course, the UNTs are due to the fact that $p_{L=100}^P > p_M^C$. The next pair of columns indicates the overall number of SNTs that have erroneously been identified as UNTs by the MLP and the GR NNs. The subsequent column pair depicts the overall number of UNTs that have wrongly been identified as SNTs by the two NNs. The next two pairs of columns present the SNT and the UNT classification error (%) demonstrated by the two NNs. Finally, the last pair of columns in Table 2 illustrates the overall NT classification error, i.e. the overall percentage of both SNTs identified as UNTs and UNTs identified as SNTs by the two NNs.

Table 2. Comparative results concerning the correct or faulty detection of successful negotiation threads (SNTs), as well as unsuccessful negotiation threads (UNTs) by the two NN-assisted negotiation strategies, when the PA adopts a hybrid negotiation strategy. Each table row represents one experiment set where the PA adopts a specific strategy coupling indicated in the first table column.

PA Hybrid Strategy Coupling	#SNTs#UNTs		#SNTs classified as UNTs		#UNTs' classified as SNTs		SNT classification error (%)		UNT classification error (%)		Overall NT classification error (%)	
			MLP	GR	MLP	GR	MLP	GR	MLP	GR	MLP	GR
	Poly&RTFT	1505	145	117	21	1	22	7.8	1.4	0.7	15.2	7.2
Poly&RATFT	1505	145	141	26	0	13	9.4	1.7	0	9.0	8.5	2.4
Poly&ATFT	1505	145	145	23	0	22	9.6	1.5	0	15.2	8.8	2.7
Exp&RTFT	873	777	19	10	0	237	2.2	1.1	0	30.5	1.2	15.0
Exp&RATFT	873	777	38	10	75	237	4.4	1.1	9.7	30.5	6.8	15.0
Exp&ATFT	873	777	48	10	0	237	5.5	1.1	0	30.5	2.9	15.0
OVERALL							7.1	1.4	2.7	27.8	5.9	8.8

As indicated in the results presented in Table 2, the GR NN manages to correctly classify most of the SNT's, incorrectly detecting as UNT's only 1.4% of the SNTs in average, thus outperforming the MLP NN that demonstrates 7.1% average SNT classification error. Thus, the MLP NN erroneously classifies five times more SNTs as UNTs, with respect to the GR NN, therefore eliminating the chance of the CA to reach a beneficial agreement. However, with respect to the UNT classification error,

the MLP NN greatly outperforms the GR NN, as it incorrectly detects as SNTs only 2.7% of the UNTs in average, while the respective error of the GR NN is 27.8%, i.e. ten times as much. The overall negotiation thread classification error is 5.9% for the MLP NN and 8.8% for the GR NN. Thus, in case the CA equally treasures the correct classification of both SNTs and UNTs, then the MLP-assisted strategy clearly outperforms the GR-assisted negotiation strategy, as the latter demonstrates approximately 50% higher overall classification error.

For the reasons above, it is concluded that MLP NNs are more appropriate for assisting negotiating intelligent agents to detect unsuccessful negotiation threads at an early negotiation round, in case their opponents follow hybrid strategies that consist of time-dependent and behaviour-dependent components.

6 Related Work

Neural networks have been used by various researchers in the AI domain. However, only a few have used them in automated negotiations and none for hybrid negotiation strategies or early avoidance of unsuccessful negotiation threads. In [15][16][17][18], neural network approximators are used in bilateral single-instance negotiations. Nevertheless, the experimental evaluation analysis provided to support the results of these efforts is based on significantly limited experiment datasets and mainly aim to predict the next offer of the opponent. Moreover, most of the neural networks adopted are not properly sized for online procedures and this is a significant drawback, which is sufficiently dealt with in the framework proposed in this paper, as well as in the approach presented in [18]. Over the aforementioned efforts [15][16][17][18], the approach proposed herewith presents the following advantages: it is quite lightweight and suitable for online negotiations, it requires no prior knowledge for the behaviour of the opponent, it eliminates the unsuccessful negotiation threads very early in the process thus saving the negotiators significant resources and giving them extra time to reach agreements with other parties, and finally it is applicable when opponents adopt arbitrary combinations of the basic strategies. To the best of the author's knowledge, this problem has not been dealt with by other research groups, and no similar solution has been proposed in the literature neither for simple nor hybrid strategies.

The authors have recently designed and evaluated several single-issue bilateral negotiation approaches, where the CA is enhanced by NNs. More specifically, in [19], the CA uses a lightweight feedforward back-propagation NN coupled with a fair relative tit-for-tat imitative tactic, and attempts to estimate the PA's price offer upon the expiration of the CA's deadline. This approach increases the number of agreements reached by one third in average. In [20], the performance of MLP and RBF NNs towards the prediction of the PA's offers at the last round has been compared. The experiments indicate that the number of agreements is increased by ~38% in average via both the MLP- and the RBF-assisted strategies. Nevertheless, the overall time and the number of neurons required by the MLP are considerably higher than these required by the RBF. On the other hand, and in order to early predict the potential unsuccessful negotiation threads, MLP and GR NNs have been employed by the CA in [21]. It has been observed that the MLP NN detects more than 90% of UNTs in average, outperforming by little the GR NN. However, the case of opponents adopting hybrid

strategies, which is the main focus of this paper, has not been dealt with in [21] or in other efforts of the authors.

7 Conclusions and Future Plans

The extensive experiments conducted indicate that when intelligent agents are assisted by NNs in single issue bilateral negotiations, they are capable of predicting their opponent's behaviour with significant accuracy, in case the latter adopts a hybrid negotiation strategy. Thus, CAs are enhanced with the ability to predict the potential outcome of negotiation threads and substantially reduce the duration of unsuccessful negotiation threads. For the purpose above, two NN architectures have been used for assisting CAs (i.e. MLP and GR NNs), while the PA has been assumed to negotiate based on a hybrid strategy consisting of time-dependent and behaviour-dependent components. In this framework, MLP NNs turn out to be more appropriate than the GR NNs for detecting unsuccessful negotiation threads at an early negotiation round, as they manage to identify 97.3% of them in average, thus reducing their duration by half. The MLP-assisted strategy clearly outperforms the GR-assisted negotiation strategy also concerning the overall classification error of both successful and unsuccessful negotiation threads, as the GR NNs demonstrates approximately 50% higher overall classification error than the MLP NNs in average. Thus, by exploiting properly trained MLP NNs, the CA is enhanced with the ability to avoid a possible unprofitable or even unachievable agreement. This leads to minimization of the required resources and maximization of the CAs overall profit from a series of threads for a single commodity.

After these promising results, the authors are now working on enhancing the proposed approach to support the selection of the most appropriate learning technique based on the prediction of the potential client utility achievable, which will be expressed as a function of both the minimum price offer estimation and the estimated round of agreement. Furthermore, the design of an alternative hybrid CA strategy is currently being studied, which couples the NN estimations with legacy strategies from the very first round. Additionally, it lies among the authors' future plans to consider cases where the opponent agent is not trustworthy, thus being malicious or credulous or skeptical. Finally, the authors aim to extend the proposed approach to also support multi-issue multi-lateral negotiations, to study scalability aspects and to apply the proposed techniques on PAs following arbitrary negotiation strategies, a highly challenging task in the automated negotiation research field.

References

1. Kraus, S.: Automated Negotiation and Decision Making in Multiagent Environments. In: Luck, M., Mařík, V., Štěpánková, O., Trappl, R. (eds.) ACAI 2001 and EASSS 2001. LNCS (LNAI), vol. 2086, pp. 150–172. Springer, Heidelberg (2001)
2. Fasli, M., Shehory, O. (eds.): TADA/AMEC 2006. LNCS (LNAI), vol. 4452. Springer, Heidelberg (2007)
3. Lomuscio, A., Wooldridge, M., Jennings, N.: A Classification Scheme for Negotiation in Electronic Commerce. *J. of Group Decision and Negotiation* 12(1), 31–56 (2003)

4. Rosenschein, J., Zlotkin, G.: *Rules of Encounter: Designing Conventions for Automated Negotiation among Computers*. MIT Press, Cambridge (1994)
5. Muller, H.: Negotiation principles. In: O'Hare, G., Jennings, N. (eds.) *Foundations of Distributed Artificial Intelligence*, pp. 211–229. John Wiley & Sons, New York (1996)
6. Li, H., Su, S., Lam, H.: On Automated E-Business Negotiations: Goal, Policy, Strategy, and Plans of Decision and Action. *J. of Organizational Computing and Electronic Commerce* 16(1), 1–29 (2006)
7. Faratin, P., Sierra, C., Jennings, N.J.: Negotiation Decision Functions for Autonomous Agents. *Int. J. of Robotics and Autonomous Systems* 24(3–4), 159–182 (1998)
8. Fei, Y., Wang, G.: Research on an Intelligent On-Line Negotiation System. In: *IEEE International Conference on e-Business Engineering*, pp. 722–728. IEEE Press, New York (2005)
9. Cranor, L.F., Resnick, P.: Protocols for Automated Negotiations with Buyer Anonymity and Seller Reputations. *NETNOMICS Journal* 2(1), 1–23 (2000)
10. Brzostowski, J., Kowalczyk, R.: Predicting Partner's Behaviour in Agent Negotiation. In: *5th International Joint Conference on Autonomous Agents and Multi-Agent Systems, Session Argumentation and Negotiation*, pp. 355–361. ACM, New York (2006)
11. Brzostowski, J., Kowalczyk, R.: Adaptive Negotiation with Online Prediction of Opponent Behaviour in Agent-based Negotiation. In: *2006 IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pp. 263–269. IEEE Computer Society, Washington (2006)
12. Raiffa, H.: *The Art and Science of Negotiation*. Harvard University Press, Cambridge (1982)
13. Pruitt, D.: *Negotiation Behavior*. Academic Press Inc., New York (1981)
14. Haykin, S.: *Neural Networks: A Comprehensive Foundation*, 2nd edn. Prentice Hall, London (1999)
15. Rau, H., Tsai, M., Chen, C., Shiang, W.: Learning-based Automated Negotiation between Shipper and Forwarder. *J. of Computers and Industrial Engineering* 51(3), 464–481 (2006)
16. Carbonneau, R., Kersten, G., Vahidov, R.: Predicting Opponent's Moves in Electronic Negotiations Using Neural Networks. In: *International Conference of Group Decision and Negotiation (GDN 2006)*, Karlsruhe, Germany (2006)
17. Zeng, Z.M., Meng, B., Zeng, Y.Y.: An Adaptive Learning Method in Automated Negotiation. In: Yeung, D.S., Liu, Z.-Q., Wang, X.-Z., Yan, H. (eds.) *ICMLC 2005. LNCS (LNAI)*, vol. 3930. Springer, Heidelberg (2006)
18. Oprea, M.: The Use of Adaptive Negotiation in Agent-Mediated Electronic Commerce. In: Mařík, V., Müller, J.P., Pěchouček, M. (eds.) *CEEMAS 2003. LNCS (LNAI)*, vol. 2691, pp. 594–605. Springer, Heidelberg (2003)
19. Roussaki, I., Papaioannou, I., Anagnostou, M.: Employing Neural Networks to Assist Negotiating Intelligent Agents. In: *2nd IEEE International Conference on Intelligent Environments 2006 (IE 2006)*, Athens, Greece (2006)
20. Papaioannou, I., Roussaki, I., Anagnostou, M.: Comparing the Performance of MLP and RBF Neural Networks Employed by Negotiating Intelligent Agents. In: *IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT 2006)*, Hong Kong, China (2006)
21. Roussaki, I., Papaioannou, I., Anagnostou, M.: Building Automated Negotiation Strategies Enhanced by MLP and GR Neural Networks for Opponent Agent Behaviour Prognosis. In: Sandoval, F., Gonzalez Prieto, A., Cabestany, J., Graña, M. (eds.) *IWANN 2007. LNCS*, vol. 4507, pp. 152–161. Springer, Heidelberg (2007)

Electromagnetism-Like Mechanism Based Algorithm for Neural Network Training

Xiao-Juan Wang, Liang Gao, and Chao-Yong Zhang

State Key Laboratory of Digital Manufacturing Equipment and Technology Department of Industrial & Manufacturing Systems Engineering, Huazhong Univ. of Sci. & Tech., Wuhan, 430074, China

Wangxiaojuan521@gmail.com

Abstract. Due to the complex nature of training neural network (NN), this problem has gained popularity in the nonlinear optimization field. In order to avoid falling into local minimum because of inappropriate initial weights, a number of global search techniques are developed. This paper applies a novel global algorithm, which is electromagnetism-like mechanism (EM) algorithm, to train NN and the EM based algorithm for neural network training is presented. The performance of the proposed algorithm is evaluated in classification problems and the comparison with BP and GA algorithms shows its effectiveness.

1 Introduction

An artificial neural network (NN) is a parallel information processing system which is composed of a large number of neurons interconnected by weighted links. NNs have great ability of self-learning, self-organizing, error-tolerance and nonlinear modeling, and NNs are suited for solving classification problems. The ability of a NN to classify accurately greatly depends upon the selection of proper weights during the NN training.

Back-propagation (BP) and some variations are common for NN training. However, this kind of algorithm suffers from the typical handicaps of all gradient techniques: very slow convergence rate, converging locally and the learning parameters need to be predetermined[1].

In recent years, many improved learning algorithm are proposed to overcome the handicaps of gradient techniques. The most common of these include a direct optimization method using a polytope algorithm[2] and global search techniques like evolutionary programming[3], simulated annealing[3] and genetic algorithm (GA)[4].

Electromagnetism-like mechanism (EM) algorithm, first proposed by Birbil and Fang in 2003[5], is a global search procedure that searches from one population of points to the next population, focusing on the area of the best solution to that point. In this paper, a new algorithm based on EM for NN training is presented. Then we test the performance of EM in training NN for classification problems by comparing with BP and GA.

The paper is organized as follows: Section 2 gives the general scheme of EM. A training algorithm based on EM for NN is given in Section 3. Section 4 presents the

simulation results for the classification problems. Finally in section 5 we present some conclusions and perspective remarks.

2 Electromagnetism-Like Mechanism (EM)

EM optimization heuristic is first proposed by Birbil and Fang (2003) for global optimization problems. It simulates the behavior of electrically charged particles and utilizes an attraction-repulsion mechanism to move a population of points toward optimality. Compared with other well-known optimization methods, EM has shown a substantial performance. It is also proved to exhibit global convergence with probability one [6], [7].

It makes all the points converge to the highly attractive valley and move further away from the steeper hills simultaneously. In EM, each point is treated as a charged particle and the charge is determined by the objective function value. The better the point is, the less charge it loads. And it is more likely to attract the other points. After calculation of all the charges, the direction of the movement for each point is determined by the total force exerted on it.

This paper introduces the procedures of EM on the nonlinear optimization problems with bounded variables in the following form:

$\min f(x)$, subject to $x \in S$. where $f(x)$ is a nonlinear function and $S = \{x \in R^n \mid -\infty < l_k \leq x_k \leq u_k < \infty, k = 1, \dots, n\}$ is a bounded feasible region. The parameters are dealt with as follows. n is the dimension of the problem. u_k is the upper bound of the k^{th} dimension. l_k is the lower bound of the k^{th} dimension.

EM consists of four phases: initialization, local search, calculation of the total force exerted on each point and movement along the direction of the force.

(1) Initialization: m points are sampled from the feasible domain. Each coordinate of the point is uniformly distributed between corresponding upper bound and lower bound. Then we can calculate the objective function value after each point is sampled. When the m points are all identified, the point with the best objective function value is stored into x_{best} .

(2) Local search: This procedure is used to gather the local information for a point. For EM, local search is very important for both the ability of exploration and exploitation.

(3) Calculation of the total force: This procedure is the most important one in the whole scheme of EM for balancing the searching time and searching quality.

The charges of the points are computed according to their objective function values. The charge q_i is calculated as follows:

$$q_i = \exp \left(-n \frac{f(x_i) - f(x_{best})}{\sum_{k=1}^m (f(x_k) - f(x_{best}))} \right), i = 1, 2, \dots, m \quad (1)$$

Then, the total force F_i exerted on point i can be worked out as follows:

$$F_i = \sum_{j \neq i}^m \begin{cases} (x_j - x_i) \frac{q_i q_j}{\|x_j - x_i\|^2} & \text{if } f(x_j) < f(x_i) \\ (x_i - x_j) \frac{q_i q_j}{\|x_j - x_i\|^2} & \text{if } f(x_j) \geq f(x_i) \end{cases} \quad i = 1, 2, \dots, m \quad (2)$$

As we can see, between two points, the point having the better objective function value attracts the other one, on the contrary, the point having the worse objective function value repulses the other one.

(4) Movement of the particles: After calculating the total force vector F_i , the point i is moved in the direction of the force by a random step length as given as follows:

$$x_i = x_i + \lambda \frac{F_i}{\|F_i\|} (RNG), i = 1, 2, \dots, m \quad (3)$$

λ is the random step length assumed to be uniformly distributed between 0 and 1. We select a random step length to ensure that the points have a nonzero probability to move to the unvisited regions along the direction of the force. RNG is a vector denoting the allowed feasible movement toward the upper bound or the lower bound for the corresponding dimension.

3 EM Based Algorithm for NN Training

The architecture of the neural network is chosen to be a three-layer fully connected feedforward network. In general, before training, a number of samples for training are chosen. For each sample, there must be some error between the real outputs and the expected outputs. In the training algorithm, the objective function is defined as the sum of squared error. The training procedure is used to find the proper weights by minimizing the objective function. The sum of squared error is defined as:

$$E = \frac{1}{2C} \sum_{i=1}^N \sum_{j=1}^C (y_{j,i} - d_{j,i})^2 \quad (4)$$

where N is the number of samples in training set. C is the number of nodes of the output layer. $y_{j,i}$ is the real output value from the j^{th} output of the i^{th} sample. $d_{j,i}$ is the expected output value from the j^{th} output of the i^{th} sample.

In the training algorithm, each point represents a string including all the weights and thresholds. The charge of each point is determined according to the network error in the current weights and thresholds. On the basis of the procedures of EM, the steps of the EM based algorithm for NN training are described as followings:

Step1: Initialization. Choose an initial population N and the maximum number of iteration, all the initial strings of weights and thresholds are generated at random in a definite range.

Step2: Calculate the error. In the case of the current weights and thresholds, compute the real output of the network for each sample and calculate the error E according to the equation (4).

Step3: Calculate q_i and F_i . Set the objective function value $f(x)$ with E , the objective of training is to minimize it. Then calculate the values of q_i and F_i according to the equations (1) and (2).

Step4: Update the variables. The set of connection weights and thresholds is just as the variable x in the objective function $f(x)$, thus update the variables according to equation (3).

Step5: Stop if the maximum number of iteration is reached, if not, go back to Step2.

4 Simulation Results and Analysis

In this study, three of benchmark classification problems are chosen. These problems include Iris, Ionosphere and Breast cancer.

Iris problem is used to identify the category of Iris according to four properties of one flower. This problem set consists of 150 examples, all of these can be classified into three categories and each category accounts for one third. The examples are split with 101 for training and 49 for testing.

Ionosphere problem is used to classify the radar returns from the ionosphere. This problem set consists of 351 examples. Each example includes 34 inputs which are continuous real number representing the parameters of electromagnetic signals. The outputs are classified into two categories good and bad. The examples are split with 200 for training and 151 for testing.

Breast cancer problem is used to correctly diagnose breast lumps as either benign or malignant based on data from examination of cells. This problem set consists of 699 examples. Each example includes 9 parameters from examination for input and 2 outputs. The examples are split with 500 for training and 199 for testing.

In order to evaluate the performance of the proposed algorithm, we compare the computational results with that of BP and GA. In the BP algorithm, the learning rate η is set 0.7 and the momentum factor μ is set 0.8. In GA, the population number n is set 80, the crossover probability p_c is set 0.9 and the mutation probability p_m is set 0.01. In EM, the initial weights are random numbers in the range $[-1, 1]$.

The computational results are shown in Table 1. The results show EM performs better than BP algorithm and GA in terms of classification accuracy and convergence speed. It can be concluded that when the training time is almost the same, the performance of the proposed algorithm will be better in classification accuracy; and when the same criterion of error is reached, EM will need less computational expense.

The number of hidden nodes and the maximum number of iterations is also demonstrated in Table 1. In the Iris problem, when the maximum number of iterations is set to 400, the classification accuracy of EM showed no superiority than GA. So we increase the maximum number of iterations to 500, then the classification accuracy is improved and the computational time increases a little. We selected the best results in 20 runs, and the CPU time is the average time of the 20 runs. All the data about GA and BP is taken from the reference [8].

Table 1. Comparison of EM, GA and BP

		Iris	Iono- sphere	Breast can- cer
E M	Classification error	1.771	3.369	0.477
	Classification accuracy	97.959%	96.689%	98.851%
	CPU time(s)	45.203	178.141	82.953
	Number of hidden nodes	12	8	8
	Maximum number of iterations	500	800	400
G A	Classification error	2.928	4.369	1.499
	Classification accuracy	93.877%	93.377%	98.492%
	CPU time(s)	48.369	817.596	286.782
	Number of hidden nodes	8	15	12
	Maximum number of iterations	400	800	400
B P	Classification error	2.198	4.037	1.556
	Classification accuracy	95.918%	93.377%	98.492%
	CPU time(s)	55.340	811.477	382.580
	Number of hidden nodes	8	15	12
	Maximum number of iterations	1000	2000	1000

5 Conclusion

A novel neural network training algorithm based on EM is proposed in this paper. Experiment results show that EM, as a powerful global optimization method, is successful in training neural networks for classification problem. In the comparison with BP and GA, the proposed algorithm is more efficient in terms of classification accuracy and computational expense.

There are still some directions to be followed. Firstly, EM can be used to train the structure of NN in order to tune the links. Moreover, combination with other local search scheme can enhance the efficiency of EM. It is also promising to apply the trained neural network in more real-word applications.

Acknowledgments. The authors would like to acknowledge the support of the National High-Tech. R&D Program, China (No. 2006AA04Z131) and the National Natural Science Foundation, China (No.50305008). The opinions and conclusions presented in this paper are those of the authors and do not necessarily reflect the views of the sponsoring agencies.

References

1. Aouiti, C., Alimi, A.M., Karray, F., Maalej, A.: A Hierarchical Genetic Algorithm for the Design of Beta Basis Function Neural Network. In: Proceedings of the 2002 International Joint Conference on Neural Networks, pp. 1246–1251 (2002)
2. Curry, B., Morgan, P.: Neural Networks: A Need for Caution, OMEGA. *International Journal of Management Sciences* 25, 123–133 (1997)
3. Salchenberger, L.M., Cinar, E.M., Lash, N.A.: Neural Networks: A New Tool for Predicting Thrift Failures. *Decision Sciences* 23, 899–916 (1992)
4. Sexton, R.S., Dorsey, R.E., Johnson, J.D.: Toward Global Optimization of Neural Networks: A Comparison of the Genetic Algorithm and Backpropagation. *Decision Support Systems* 22, 171–186 (1998)
5. Birbil, Ş.İ., Fang, S.C.: An Electromagnetism-like Mechanism for Global Optimization. *Journal of Global Optimization* 25, 263–282 (2003)
6. Birbil, Ş.İ., Fang, S.C.: A Multi-point Stochastic Search Method for Global Optimization. In: Proceedings of the 4th International Symposium, ISOR 2002 (2002)
7. Birbil, Ş.İ.: Stochastic Global Optimization Techniques. Ph.D. Thesis, North Carolina State University, Raleigh, NC, USA (2002)
8. Gao, H.B., Gao, L., Zhou, C.: Particle Swarm Optimization Based Algorithm for Neural Network Learning. *Acta Electronica Sinica* 9, 1572–1579 (2004)

Evolvable Neuro-fuzzy System for Artificial Creativity in Linguistics

Keith Douglas Stuart¹ and Maciej Majewski²

¹ Polytechnic University of Valencia, Department of Applied Linguistics
Camino de Vera, s/n, 46022 Valencia, Spain

kstuart@idm.upv.es

² Koszalin University of Technology, Faculty of Mechanical Engineering
Raclawicka 15-17, 75-620 Koszalin, Poland

maciej.majewski@tu.koszalin.pl

Abstract. The paper describes an application of evolvable fuzzy neural networks for artificial creativity in linguistics, which consists of the intelligent mechanisms of affix and root morpheme creativity and analysis. The task of the creation of an English vocabulary was resolved with neural networks which have an evolvable architecture with learning capabilities as well as a fuzzy connectionist structure. The paper features a form of artificial creativity which creates words on its own using genetic algorithms, fuzzy logic methods, and multiple layer neural networks. A review of selected issues is carried out with regards to linguistic creativity as well as spontaneous language production. The paper also presents experimental results of affix and root morpheme recognition in the context of word meaning analysis.

Keywords: artificial intelligence; artificial creativity; evolvable fuzzy neural networks; natural language processing.

1 Artificial Creativity in Linguistics

The paper deals with the most productive word formation processes of the English language, i.e. derivation, which includes prefixation, suffixation and infixation. Artificial creativity in linguistics presented in the paper involves creation of new words using evolvable fuzzy neural networks. This artificial creativeness is a process involving the generation of new morphemes or words, or new associations between existing morphemes or words. The initialization of the artificial creativeness proposed in the system replaces primitively some functions of the frontal lobes which have been found to play a part in spontaneous language production.

We are speaking here only metaphorically as we do not know the nature of the neurobiological implementation of the functions of language in the form of electrophysiological responses in the brain. However, it seems clear that neural networks are a good working metaphor for the functions of language in the brain [1]. We start from the hypothesis of a neural basis for language.

The approach we are using here is that of hybrid neural networks towards the creation of an English vocabulary based on learning patterns created with lists made of essential words or limited vocabularies. The motivation for using binary neural networks for the creation of an English vocabulary is that they offer the advantage of simple binarization of morphemes and words, as well as very fast training and run-time response of this type of neural networks [2,3,4].

2 The State of the Art

According to the literature, the main reason for investing time, research budget, and talent in the quest for a truly creative machine is to work towards an answer to the question of whether machines can be made to behave like humans. If the creative side of human cognition can be captured by computation, then it is very likely that computers can acquire human-like skills and capabilities.

In the light of recent research work, it is believed that, as humans uncover reasons for thinking that human creativity is in fact beyond the reach of computation, they will be inspired to nonetheless engineer systems that dodge these reasons and appear to be creative. A side effect of the human approach is perhaps to provide artificial intelligence with at least some early steps towards a theoretical foundation for machine creativity. Absent such a foundation which to be effective, would presumably have to be somewhat mathematical in nature, artificial creative agents will never be achieved.

3 Description of the Method

The artificial limbic system of the human brain is represented in the system primitively with affective computing software which involves tasks related to the frontal lobes activity in spontaneous language production. Fuzzy logic description of language production is associated with memory which consists of encoded knowledge base. The activation values of language production are the inputs of the artificial neural network of a type of neuronal classifier. The threshold value of the output of the network can be half of the maximum value. The classifier initializes binary images of graphemes. Each of the output values which are

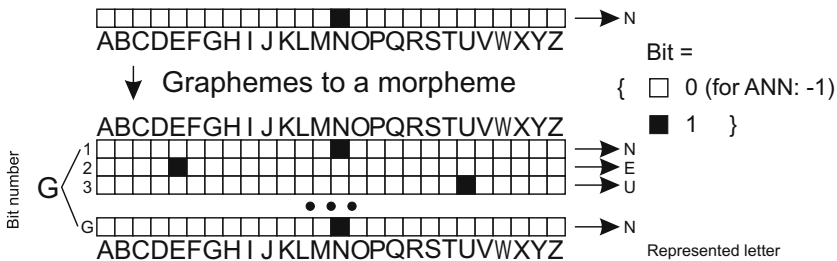


Fig. 1. Binary image of a grapheme and morpheme

greater than the threshold value indicates the bit representing a specified letter in the binary image. The initialized graphemes are consolidated into a morpheme represented as a two dimensional binary image (Fig. 1). The grapheme number parameter which determines the size of the morpheme as a binary image is a chromosome for the evolutionary algorithms. That chromosome is a subject for evolutionary optimization (Fig. 2a) with the fitness function based on statistics of the number of graphemes of affixes (prefixes, suffixes and infixes). The initialized binary image of a morpheme determines the evolvable architecture of the artificial neural network for recognizing affixes. The evolvable network is a 3 layer Hamming net with a Maxnet layer (Fig. 3). Fuzzy modeling of learning patterns is based on restrictions on the formation of affixes. Learning is also supported with evolvable affixes, which are the generations of chromosomes

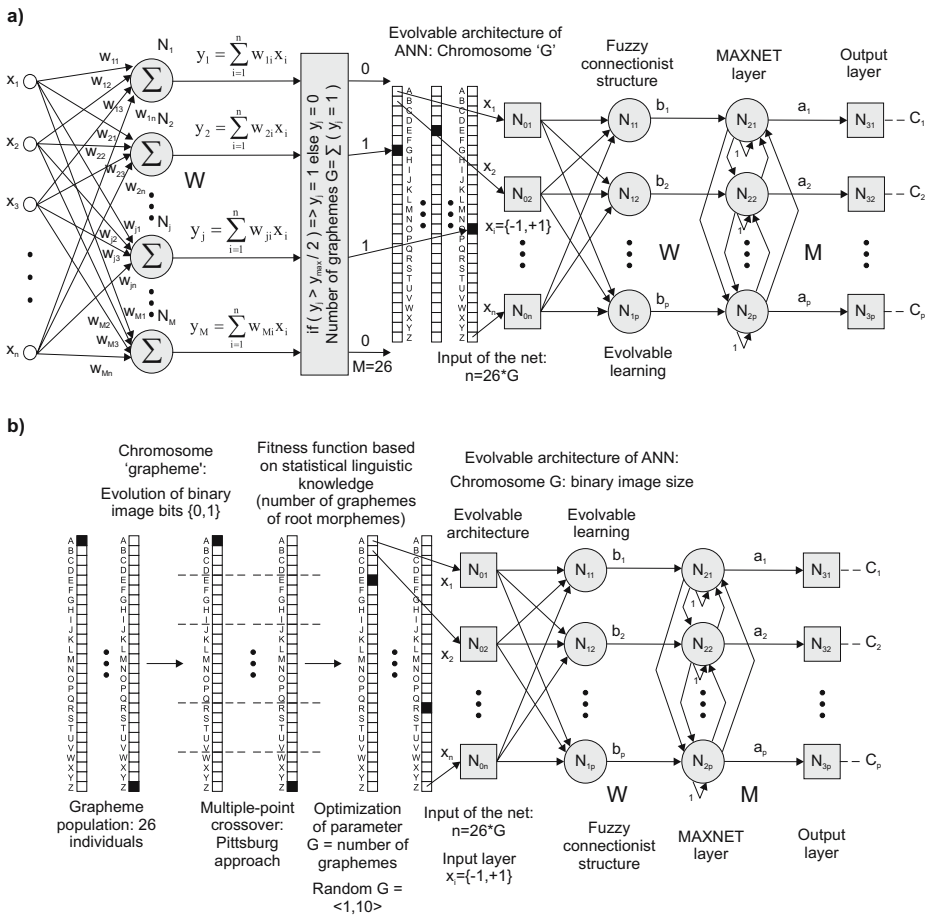


Fig. 2. Illustration of (a) the problem of the morpheme (affix) creativity, (b) the problem of the root morpheme creativity using evolvable fuzzy neural networks

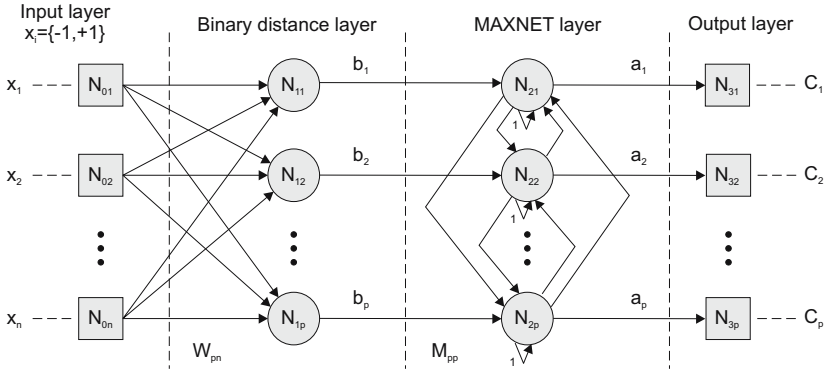


Fig. 3. Structure of the Hamming neural network

prefix, suffix and infix. The fitness function of the evolutionary optimization of the affixes is based on fuzzy restrictions on affix formation. The output neurons of the 3 layer neural network represent the classes of affixes. The network recognizes the affix by directly applying the one-nearest-neighbor classification rule. The grapheme population of 26 individuals represents each letter of the modern Latin alphabet. The chromosome which determines the size of a binary image of a morpheme is a subject for evolutionary optimization with the fitness function based on statistics of the number of graphemes of root morphemes. The grapheme population is a subject for multi-point crossover type evolution of its binary image bits [5,6,7]. The initialized binary image of the evolvable morpheme determines the evolvable architecture of the 3-layer artificial neural network for recognizing root morphemes. The connectionist structure of the network is determined by the evolutionary morpheme binary image which is evolvable in terms of its size and bit values (Fig. 2b). The network is equipped with learning patterns of restrictions on the formation of root morphemes using fuzzy modeling. The fuzzy reasoning is applied for the learning and connectionist structure. The fuzzy learning patterns are evolving chromosomes with regard to image bits. The network recognizes root morphemes as word components represented by output neurons with classification into root morpheme classes. The construction of a new word is preceded by prefix derivations, suffix derivations, the root morpheme and infix insertion. The word formation consists of consolidation of the created morphemes with respect to the proper word structure composed of prefixes, root morphemes, infixes and suffixes.

The cycle of artificial creativity in linguistics using evolvable fuzzy neural networks is shown in Fig. 4. The hybrid system uses neural classifiers equipped with evolvable fuzzy logic membership functions and rules (Fig. 5). The structure of the neural network for morpheme creation, which consists of input and hidden neurons, is shown in Fig. 6.

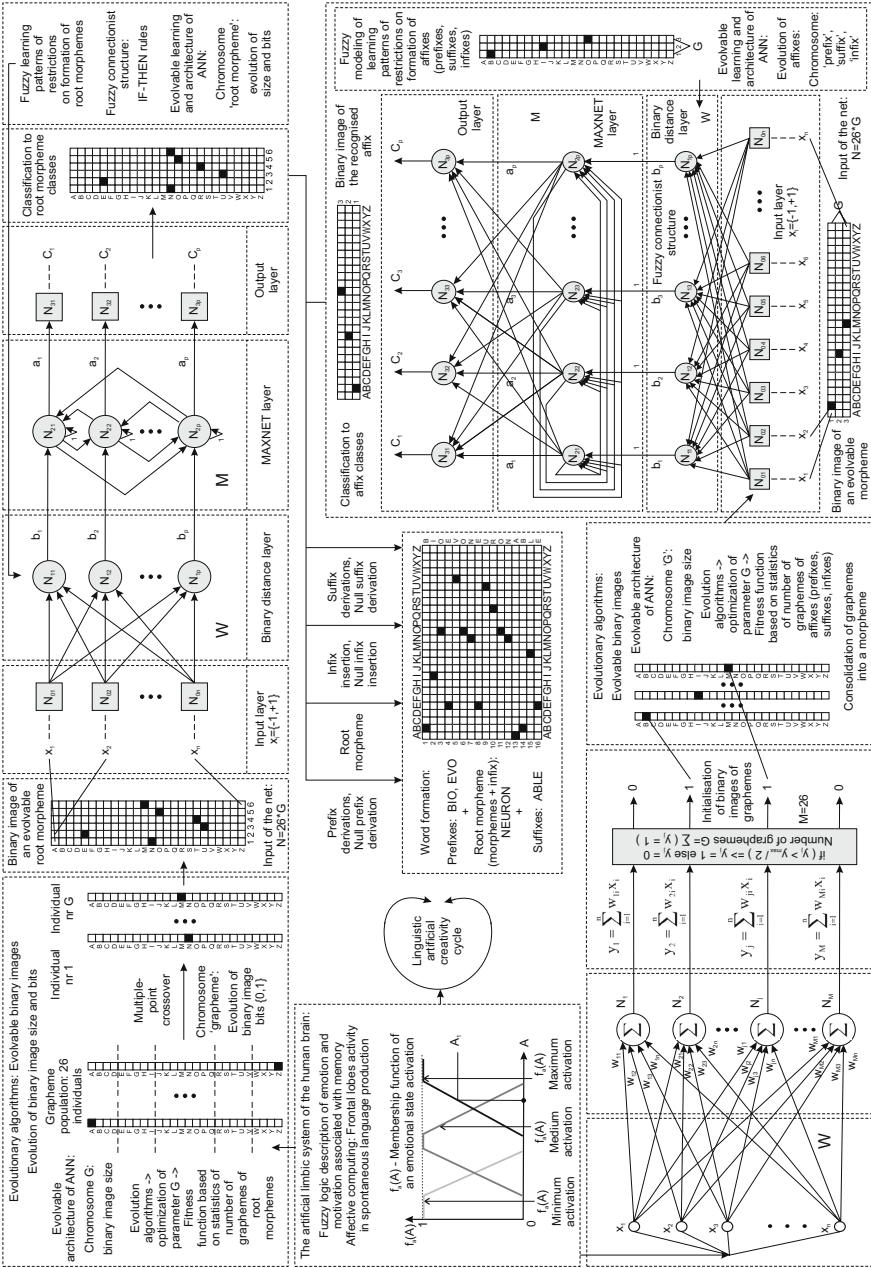


Fig. 4. Cycle of artificial creativity in linguistics using evolvable fuzzy neural networks

4 Experimental Results

The results of the experiments with the implementation of this hybrid system architecture were evaluated. The system was also examined with regard to the repeatability of the results.

As shown in Fig. 7, the ability of the implemented evolvable fuzzy neural networks to recognise an affix and root morpheme depends on the number of letters of that letter string. For best performance, the neural network requires a minimum number of letters of each morpheme being recognized as its input.

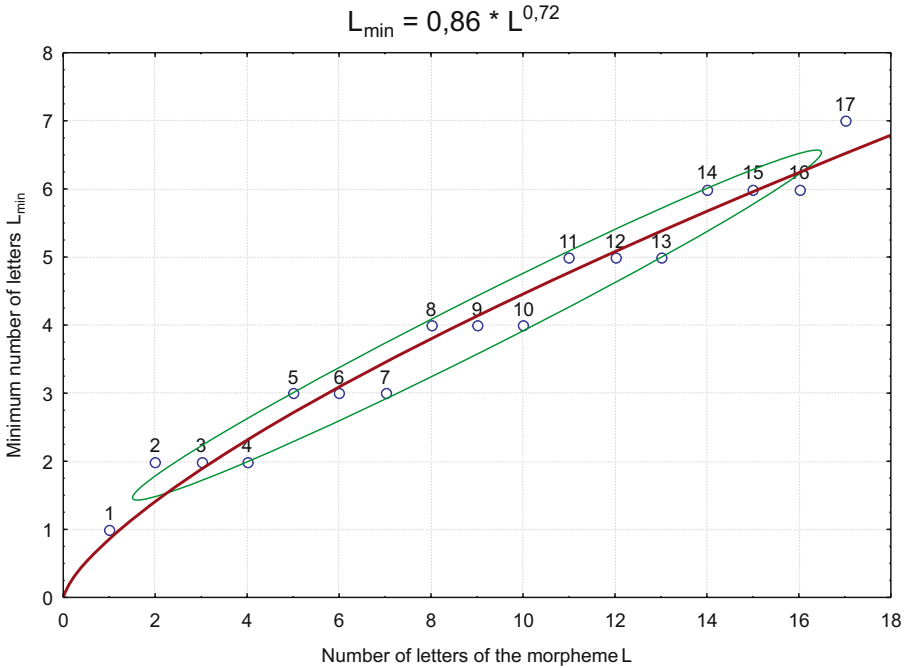


Fig. 7. Sensitivity of morpheme recognition: minimum number of letters of the morpheme being recognized vs. number of morpheme letters

5 Conclusions and Perspectives

Creativity is generally regarded to involve breaking the kind of rigid rules standing at the heart of logic. However, the proposed architecture provides a form of creativity that does not defy logic. It is believed that a good artificial vocabulary generator may provide a wide variability. The experimental results showed that the proposed artificial intelligence system architecture can secure wide variability.

Application of binary evolvable fuzzy neural networks allows for recognition of morphemes with similar meanings but different letter string patterns. The method presented in this paper can be easily extended.

In the literature there are very few reports about the discussed problem in the field of artificial creativity in linguistics. The method proposed in this paper is a conceptually new approach to this problem. The experimental results of the proposed method of creation of an English vocabulary show its promising performance, and can be used for further development and experiments.

In the future, natural language will undoubtedly be the most common and direct way for communication between humans and computers. The proposed evolvable fuzzy neural networks are both effective and flexible which makes their application possible in many fields of natural language processing.

References

1. Taylor, N., Taylor, J.: The Neural Networks for Language in the Brain: Creating LAD. In: Computational Models for Neuroscience, pp. 245–265. Springer, London (2003)
2. Kacalak, W., Stuart, K., Majewski, M.: Intelligent Natural Language Processing. In: Jiao, L., Wang, L., Gao, X.-b., Liu, J., Wu, F. (eds.) ICNC 2006. LNCS, vol. 4221, pp. 584–587. Springer, Heidelberg (2006)
3. Stuart, K., Majewski, M.: Selected Problems of Knowledge Discovery Using Artificial Neural Networks. In: Liu, D., Fei, S., Hou, Z., Zhang, H., Sun, C. (eds.) ISNN 2007. LNCS, vol. 4493, pp. 1049–1057. Springer, Heidelberg (2007)
4. Stuart, K., Majewski, M.: A New Method for Intelligent Knowledge Discovery. In: Melin, P., Castillo, O., Aguilar, L.T., Kacprzyk, J., Pedrycz, W. (eds.) IFSA 2007. LNCS (LNAI), vol. 4529, pp. 721–729. Springer, Heidelberg (2007)
5. Back, T., Fogel, D., Michalewicz, Z.: Handbook of Evolutionary Computation. Oxford University Press, New York (1997)
6. Goldberg, D.E.: Genetic Algorithms in Search, Optimisation and Machine Learning. Addison-Wesley, Reading (1989)
7. GALib: A C++ Library of Genetic Algorithm Components. Massachusetts Institute of Technology, Cambridge (1996), <http://lancet.mit.edu/ga/>

Extended Legendre Wavelets Neural Network

XiaoYang Zheng¹, XiaoFan Yang², Hong Su³, and Yong Wu⁴

¹ College of Mathematics and Physics, Chongqing University, Chongqing 400030, China

² College of Computer Science, Chongqing University, Chongqing 400030, China

^{3,4} College of Mathematics and Physics, Chongqing Institute of Technology,
Chongqing 400050, China

{zhengxiaoyang,wuyong}@cqit.edu.cn

Abstract. Based on analyzing Legendre wavelets, this paper presents the extended Legendre wavelets (ELW), which is defined on the interval $(-r, r)$, and proves to be orthogonal. Furthermore, this paper constructs the extended Legendre wavelet neural network (ELWNN) by using the ELW functions instead of the activation functions of a multilayer perceptron neural network (MPNN). The ELWNN has advantages of simple structure and very efficient convergence rate. Additionally, the ELWNN is applied to the approximation a function and better approximate results are achieved.

Keywords: Legendre wavelets; multilayer perceptron neural network; extended Legendre wavelets neural network.

1 Introduction

Artificial neural networks are computational tools for pattern classification but recently have been widely used to solve a wide range of problems related to signal processing, because they provide high quality of approximation, prediction of substantially stochastic and chaotic signals under a priori and current uncertainty [1]. While the Wavelet neural network is a neural network where wavelet basis functions are used as activation functions [2,3,4]. While because of the properties of Legendre wavelets, the Legendre wavelets neural network has simple architecture [5]. This paper presents the ELW. Regarding the ELW functions as the activation functions of a MPNN, the ELWNN is constructed and it is applied to the approximation of a function trained with gradient descent algorithm and better results are obtained, compared with the MPNN trained with Levenberg-Marquardt algorithm. This demonstrates the validity and applicability of the ELWNN.

2 Properties of Legendre Wavelets

In this section, Legendre wavelets are analyzed. Taking advantage of the property of Legendre wavelets presented by piecewise polynomials, this paper presents the translation property of Legendre wavelets.

Legendre wavelets are the functions for the form [7-9]

$$\psi_{nm}^k(x) = \begin{cases} \left(k + \frac{1}{2}\right)^{\frac{1}{2}} 2^{\frac{n+1}{2}} L_k(2^{n+1}x - 2m - 1) & \frac{m}{2^n} \leq x < \frac{m+1}{2^n} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where the functions $L_k(x)$ are the k degree Legendre polynomials and satisfy the recursive formula:

$$L_0(x) = 1$$

$$L_1(x) = x,$$

$$L_{k+1}(x) = \frac{2k+1}{k+1} x L_k(x) - \frac{k}{k+1} L_{k-1}(x),$$

$$k = 1, 2, 3, \dots$$

and n denotes decomposition level for $n = 1, 2, \dots$; m denotes integer translation for $m = 0, 1, \dots, 2^n - 1$, and k is the degree of the Legendre polynomials. Actually, Legendre wavelets approximate a function by piecewise Legendre polynomials. Then, Legendre wavelets that are defined on the interval $[0, 1)$ can be obtained through a translation operator transformation on Legendre wavelets, defined on the subinterval. Here, a concept of translation operator is presented.

Definition 1. A translation operator is defined by

$$T(i)f : f(x) \rightarrow g(x),$$

where the $T_{right}(i)$, $T_{left}(i)$ and $T(i)$ denote the function $g(x)$ derived from the function $f(x)$ by translating i times on the right, on the left and on the right and left, respectively.

Lemma 1. Legendre wavelets, defined on the interval $[0, 1)$, can be obtained through the $T_{right}(m)$ transformation on Legendre wavelets, defined on the subinterval $[0, \frac{1}{2^n})$, and are defined by

$$\psi_{nm}^k(x) = \begin{cases} \left(k + \frac{1}{2}\right)^{\frac{1}{2}} 2^{\frac{n+1}{2}} L_k(2^{n+1}x - 2m - 1) & \frac{m}{2^n} \leq x < \frac{m+1}{2^n} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where $m = 1, \dots, 2^n - 1$.

It is very easy to prove [10]. According to the translation property, the definition interval of Legendre wavelets can be extended. Thus, a concept of the ELW is presented.

3 Extended Legendre Wavelets and Its Properties

In this section, the definition interval of Legendre wavelets is extended to the interval $(-r, r)$. Then, properties of the ELW are analyzed. Now, if the interval $[0,1)$ is divided by a positive integer $a(a > 2, a \in N)$, then Legendre wavelets, defined on the interval $[0, \frac{1}{a^n})$, can be obtained by

$$\psi_{n0}^k(t) = \begin{cases} (2k+1)^{\frac{1}{2}} a^{\frac{n}{2}} L_k(2a^n x - 1) & 0 \leq x < \frac{1}{a^n} \\ 0 & \textit{otherwise} \end{cases} \quad (3)$$

Similarly, Legendre wavelets, defined on the interval $[0,1)$, can be obtained through the $T_{right}(m)$ and are defined by

$$\psi_{nm}^k(x) = \begin{cases} (2k+1)^{\frac{1}{2}} a^{\frac{n}{2}} L_k(2a^n x - 2m - 1) & \frac{m}{a^n} \leq x < \frac{m+1}{a^n} \\ 0 & \textit{otherwise} \end{cases} \quad (4)$$

where $m = 0, 1, \dots, a^n - 1$.

Theorem 1. Legendre wavelets, defined on the interval $(-\frac{i}{a^n}, \frac{i}{a^n}) (i \in N)$, can be obtained through the $T(i)$ transformation on Legendre wavelets defined on the sub-interval $[0, \frac{1}{a^n})$ and are defined by

$$\psi_{nm}^k(x) = \begin{cases} (2k+1)^{\frac{1}{2}} a^{\frac{n}{2}} L_k(2a^n x - 2m - 1) & \frac{m}{a^n} \leq x < \frac{m+1}{a^n} \\ 0 & \textit{otherwise} \end{cases} \quad (5)$$

where $m = -i, \dots, 0, 1, \dots, i - 1$.

Now, since for any a rational constant r , there exists two positive integer a and i such that $r = \frac{i}{a^n}$ or $r = -\frac{i}{a^n}$, then the ELW, defined on the interval $(-r, r)$, can be obtained.

Theorem 2. The extended Legendre wavelets, defined on the interval $(-\frac{i}{a^n}, \frac{i}{a^n})$, are an orthogonal set.

4 Extended Legendre Wavelets Neural Network

In this section, the simple ELWNN is constructed. The ELWNN, as shown in figure1, is a neural network with two layers. The neurons of the hidden layer in a neural network have ELW activation functions of different resolutions.

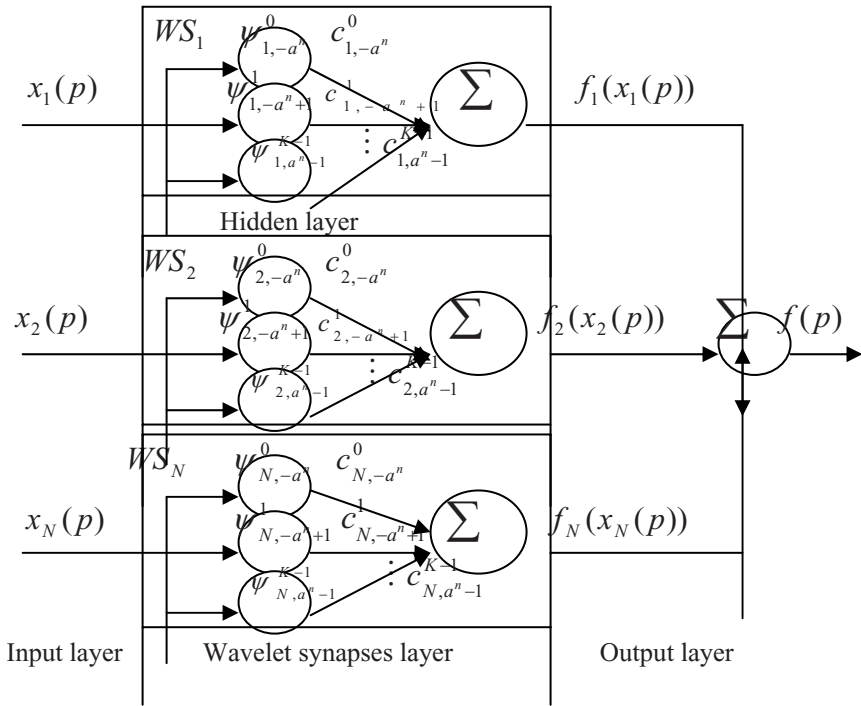


Fig. 1. The extended Legendre wavelets neural network

The ELWNN consists of an input layer, a wavelet synapses layer and an output layer with linear functions. Where $x_n(p)$ represents the n th decomposition level input variable and the $p = 0, 1, \dots$ is discrete time, WS_n denotes the n th decomposition level Legendre wavelets approximation and $f(p)$ is the actual output of the ELWNN.

5 Experimental Results

Now, let $n = 1$, $k = 3$, $m = -2, -1, 0, 1$, then there exist twelve ELW functions. It is easy to get an ELWNN through regarding the twelve ELW functions as the activation functions of the MPNN. The efficiency of the ELWNN and introduced gradient descent algorithm was tested by a function

$$y = e^{-x} \cos 5x \quad x \in (-1, 1).$$

The training data set contained 200 values for $x = -1, -0.99, \dots, 0.99$, and the testing data contained values for $x = -1, -0.97, \dots, 0.97$. The ELWNN has one synapse, corresponding to the four inputs and twelve Legendre wavelets functions synapse for the synapse, and it was trained with the gradient descent algorithm. The normalized root mean squared error measure was used to estimate the forecast accuracy on the testing data. The results are compared with those obtained using a MPNN. The MPNN has one hidden layer (containing five neurons with hyperbolic tangent activation functions) and a linear output layer. The weights of the MPNN are trained using the Levenberg-Marquardt algorithm. The summary of the two methods approximating the function is given in the table 1.

Table 1. Performance of the networks on approximating the function

Network/ Training procedure/ iterations	MSE
ELWNN/ Gradient descent/ six	9.43×10^{-4}
MPNN/ Levenberg-Marquardt/ fifteen	5.45×10^{-4}

From the table1, the ELWNN displays a shorter computing time as compared to the MPNN although trained with gradient descent algorithm while to Levenberg-Marquardt algorithm. Thus the ELWNN has an efficient rate of convergence and the smallest number of parameters and it could be efficiently used for prediction, compression and classification of various non-stationary noisy signals.

6 Conclusions

Extended Legendre wavelets is presented and a simple ELWNN is constructed. This neural network is tested by a function using the gradient descent algorithm and better approximation results are obtained. This demonstrates the validity and applicability of the ELWNN.

Acknowledgements. The work described in this paper was supported by grants from the National Natural Science Foundation of China (50573095).

References

1. Foo, S.Y., Stuart, G., Harvey, B., Meyer-Baese, A.: Neural Network-based EKG Pattern Recognition. *Engineering Applications of Artificial Intelligence* 15, 253–260 (2002)
2. Basheer, I.A., Hajmeer, M.: Artificial Neural Networks: Fundamentals, Computing, Design, and Application. *Journal of Microbiological Methods* 43, 3–30 (2000)
3. Kobayashi, K., Torioka, T.: A Wavelet Neural Network for Function Approximation and Network Optimization. In: Dagli, C.H., Fernandez, B.R., Ghosh, J., Sounder Kumara, R.T. (eds.) *Intelligent Engineering Systems Through Artificial Neural Networks*, vol. 4, pp. 505–510. ASME Press (1994)
4. Razzaghi, M., Yousefi, S.: Legendre Wavelets Method for the Solution of Nonlinear Problems in the Calculus of Variations. *Math. Comput. Modelling* 34, 53–54 (2001)
5. Yousefi, S., Razzaghi, M.: Legendre Wavelets Method for the Nonlinear Volterra-Fredholm Integral Equations. *Math. Comput. Simulation* 70, 1–5 (2005)
6. Azzaghi, M., Yousefi, S.: The Legendre Wavelets Operational Matrix of Integration. [J.] *International Journal of Systems Science* 32, 500–502 (2001)
7. Maleknejad, K., Sohrabi, S.: Numerical Solution of Fredholm Integral Equation of the First Kind by Using Legendre Wavelets. *Appl. Math. Comput.* 186, 837–843 (2007)
8. Alpert, B.: A Class of Bases in for the Sparse Representation of Integral Operators. *SIAM J. Math. Anal.* 24, 247–262 (1993)
9. Berlkin, G.: On the Representation of Operators in Bases of Compactly Supported Wavelets. *SIAM J. Numer. Anal.* 6, 1716–1739 (1992)
10. Alpert, B., Beylkin, G., Gines, D., Vozovoi, L.: Adaptive Solution Partial Differential Equations in Multiwavelet Bases. *J. Comp. Phys.* 182, 149–190 (2002)

Growing Algorithm of Laguerre Orthogonal Basis Neural Network with Weights Directly Determined

Yunong Zhang¹, Tongke Zhong¹, Wei Li¹, Xiuchun Xiao^{1,2}, and Chenfu Yi¹

¹ School of Information Science and Technology, Sun Yat-sen University,
Guangzhou 510275, China

² College of Information, Guangdong Ocean University, Zhanjiang 524088, China
zhynong@mail.sysu.edu.cn, ynzhang@ieee.org

Abstract. Determination of appropriate neural-network (NN) structure is an important issue for a given learning or training task since the NN performance depends much on it. To remedy the weakness of conventional BP neural networks and learning algorithms, a new Laguerre orthogonal basis neural network is constructed. Based on this special structure, a weights-direct-determination method is derived, which could obtain the optimal weights of such a neural network directly (or to say, just in one step). Furthermore, a growing algorithm is presented for determining immediately the smallest number of hidden-layer neurons. Theoretical analysis and simulation results substantiate the efficacy of such a Laguerre-orthogonal-basis neural network and its growing algorithm based on the weights-direct-determination method.

Keywords: Laguerre orthogonal polynomials, neural networks, weights-direct-determination, growing algorithm.

1 Introduction

Artificial neural networks (ANN) have been applied widely to many engineering and practical fields such as computational intelligence, pattern recognition, signal processing, and nonlinear control [1-5]. The conventional BP algorithm is essentially a gradient-descent optimization method, which adjusts the neural-weights to bring the network input/output behavior into a desired mapping as of some application environment. However, BP neural networks have some inherent weaknesses [6-11]. To resolve such weaknesses of BP neural networks, a new Laguerre-orthogonal-basis neural network is constructed in this paper, together with its weights-direct-determination and structure-growing method.

2 Neural-Network Model and Theoretical Analysis

The Laguerre-orthogonal-basis neural network is constructed as follows (in the first subsection), with the mathematical analysis about its approximation capability presented then (in the second subsection).

2.1 Neural Network Structure

As shown in Fig. 1, the Laguerre orthogonal basis neural network consists of three layers. The input and output of the neural network are denoted respectively as x and y . The input- and output-layers each employ one neuron with a linear activation function $f(x) = x$, while the hidden-layer has n neurons activated by a group of order-increasing Laguerre orthogonal polynomials $\varphi_j(x)$, $j = 0, 1, 2, \dots, n - 1$. Parameters w_j , $j = 0, 1, 2, \dots, n - 1$ denote the weights between the hidden-layer and output-layer neurons. Moreover, the neural-weights between the input-layer and hidden-layer neurons are fixed to be 1, and all the neuronal thresholds are fixed to be 0, which would simplify the network model and its possible circuit implementation. It is worth pointing out here that this neural network can be viewed as a special BP neural network and adopts the standard BP algorithm as its training rule.

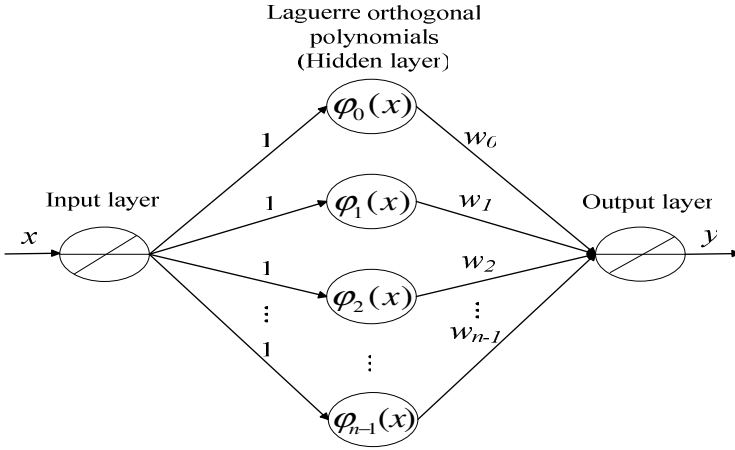


Fig. 1. Laguerre-orthogonal-basis neural network model

2.2 Theoretical Foundation

After constructing the Laguerre-orthogonal-basis neural network (actually before that), we could present some theoretical analysis on the approximation capability of the neural network. By the polynomial interpolation and curve-fitting theory [12, 13], we could always construct a polynomial function $\varphi(x)$ (i.e., an underlying input-output function of Laguerre-orthogonal-basis neural network) to approximate the unknown target function $\phi(x)$. For this approximation, we could have the following.

Definition 1. [12]. Define that $\varphi_j(x)$ is a polynomial of degree j , and that polynomial sequence $\{\varphi_j(x)\}_{j=0}^{n-1}$ is a set of n orthogonal-polynomial functions with respect to some weighting function $\rho(x)$ over a finite or infinite interval $[a, b]$; i.e., for any $i, j \in \{0, 1, 2, \dots, n - 1\}$,

$$\int_a^b \varphi_i(x)\varphi_j(x)\rho(x)dx \begin{cases} =0 & i \neq j, \\ > 0 & i = j. \end{cases}$$

Then, $\{\varphi_j(x)\}_{j=0}^{n-1}$ is termed as an orthogonal polynomial sequence with respect to weighting function $\rho(x)$ on the interval $[a, b]$.

Definition 2. [12]. Laguerre polynomials could be defined by the following equation:

$$\varphi_{n-1}(x) = e^x \frac{d^{n-1}(x^{n-1}e^{-x})}{dx^{n-1}}, \quad 0 \leq x < +\infty ,$$

which is an orthogonal polynomial of degree $n-1$ with respect to weighting function $\rho(x) = e^{-x}$ over interval $[0, +\infty)$, with its orthogonal relationship derived as below:

$$\int_0^{+\infty} \varphi_i(x)\varphi_j(x)e^{-x}dx = \begin{cases} 0 & i \neq j, \\ (i!)^2 & i = j. \end{cases}$$

Furthermore, the recurrence relation for the Laguerre orthogonal polynomials could be expressed as follows:

$$\varphi_{n-1}(x) = (2n-3-x)\varphi_{n-2}(x) - (n-2)^2\varphi_{n-3}(x), \quad n = 3, 4, \dots .$$

The first few Laguerre orthogonal polynomials can be written down readily:

$$\begin{aligned} \varphi_0(x) &= 1, \\ \varphi_1(x) &= 1 - x, \\ \varphi_2(x) &= x^2 - 4x + 2, \\ \varphi_3(x) &= -x^3 + 9x^2 - 18x + 6, \\ \varphi_4(x) &= x^4 - 16x^3 + 72x^2 - 96x + 24, \\ &\dots \dots \end{aligned}$$

From the mathematical perspective of polynomial curve-fitting, the training of the neural network is essentially the establishment of the best functional approximation, and thus we would like to present the following theoretical foundation about the approximation ability of the Laguerre-orthogonal-basis neural network.

Definition 3. [12, 13]. Assume that $\phi(x), \varphi_j(x) \in C[a, b], j = 0, 1, \dots, n-1$ (in words, target function $\phi(x)$ and every polynomial function $\varphi_j(x)$ of polynomial-function sequence $\{\varphi_j(x)\}_{j=0}^{n-1}$ are continuous over the closed interval $[a, b]$), and

that $\{\varphi_j(x)\}_{j=0}^{n-1}$ is a set of linearly independent polynomial-functions. For given weighting-function $\rho(x)$ on interval $[a, b]$, appropriate coefficients w_0, w_1, \dots, w_{n-1} could be chosen for generalized polynomial $\varphi(x) = \sum_{j=0}^{n-1} w_j \varphi_j(x)$ so as to minimize $\int_a^b (\phi(x) - \varphi(x))^2 \rho(x) dx$. Then, function $\varphi(x)$ is termed the least-square approximation of $\phi(x)$ with respect to weighting-function $\rho(x)$ over interval $[a, b]$.

Theorem 1. [12, 13]. For $\phi(x) \in C[a, b]$, its least-square approximation function $\varphi(x)$ could exist uniquely, of which the coefficients w_0, w_1, \dots, w_{n-1} are solvable.

When approximating the unknown target function $\phi(x)$, the relation between the input and output of Laguerre-orthogonal-basis neural network could become exactly

$$y = \varphi(x) = w_0 \varphi_0(x) + w_1 \varphi_1(x) + \dots + w_{n-1} \varphi_{n-1}(x) , \tag{1}$$

of which the approximation ability is guaranteed by Definitions 1 to 3 and Theorem 1.

3 Weights-Direct-Determination Method

As discussed above, weights between the hidden-layer and output-layer neurons could be generated so as to approximate effectively the unknown target function by learning (or to say, training with) the given data samples. More importantly, such neural weights between the hidden-layer and output-layer neurons could be determined directly by using the weights-direct-determination method, which generates the steady-state optimal weights just in one step without the lengthy BP iterative-training. The new method could thus remove some inherent weaknesses of conventional BP algorithms, e.g., slow convergence.

Let us take $\{(x_i, \phi_i := \phi(x_i))\}$, $i = 1, 2, \dots, m\}$ as the training (or termed, sampling) data-set, and define the batch-processing error (BPE) function E as

$$E = \frac{1}{2} \sum_{i=1}^m \left(\phi_i - \sum_{p=0}^{n-1} w_p \varphi_p(x_i) \right)^2 . \tag{2}$$

Then we have the following results about the weights-direct-determination method.

Theorem 2. The steady-state weights of the Laguerre orthogonal basis neural network can be obtained directly as (with proof omitted due to space limitation):

$$\boldsymbol{\omega} = \lim_{k \rightarrow \infty} \boldsymbol{w}(k) = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{\gamma} , \tag{3}$$

where the neural-network weights vector \mathbf{w} , the target-output vector $\boldsymbol{\gamma}$, and the input information matrix \mathbf{X} are defined respectively as

$$\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{n-1} \end{bmatrix} \in R^n, \quad \boldsymbol{\gamma} = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_m \end{bmatrix} \in R^m, \quad \mathbf{X} = \begin{bmatrix} \varphi_0(x_1) & \varphi_1(x_1) & \cdots & \varphi_{n-1}(x_1) \\ \varphi_0(x_2) & \varphi_1(x_2) & \cdots & \varphi_{n-1}(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_0(x_m) & \varphi_1(x_m) & \cdots & \varphi_{n-1}(x_m) \end{bmatrix} \in R^{m \times n}.$$

In addition, Formula (3) can be rewritten as $\boldsymbol{\omega} = \text{pinv}(\mathbf{X})\boldsymbol{\gamma}$, with $\text{pinv}(\mathbf{X})$ denoting the pseudoinverse of input information matrix \mathbf{X} , equaling $(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T$ here.

4 Growing Algorithm

Based on the aforementioned weights-direct-determination method of neural network, a growing algorithm could be developed further, which determines rapidly the optimal number of hidden-layer neurons for the constructed Laguerre neural network. The growing algorithm could be designed and stated as follows.

Step 0: Obtain training data-set $\{(x_i, \phi_i := \phi(x_i)), i = 1, 2, \dots, m\}$, with x_i and ϕ_i corresponding to the i th input- and output-values of (unknown) target function $\phi(x)$, respectively. Note that ϕ_i is also termed as expected, desired or target output.

Step 1: Construct the Laguerre neural network. Initialize the number of hidden-layer neurons to be $n = 1$ and limit the maximal number of hidden-layer neurons to be $MaxNeuNum$ (e.g., 1000). Specify the target (or desired) precision in the sense of batch-processing error (BPE) as ε (e.g., 10^{-5}).

Step 2: If $n > MaxNeuNum$, then terminate the algorithm with a notice about the inability failure. Otherwise, compute the neural-weights \mathbf{w} by using the weights-direct-determination formula (3) and calculate the actual training BPE [defined by equation (2)] at the present stage (i.e., with n hidden-layer neurons).

Step 3: If the actual training BPE $> \varepsilon$ (i.e., target precision), then update $n = n + 1$ and go back to Step 2 to continue this growing algorithm. Otherwise, terminate the algorithm by supplying the smallest number n of hidden-layer neurons, the corresponding weights \mathbf{w} and the actual training BPE.

In the above growing algorithm, the neural-weights direct-determination method is applied to computing the Laguerre network weights. Therefore, the optimal number of hidden-layer neurons can be decided rapidly and deterministically for the Laguerre orthogonal basis neural network. In the ensuing section, we would like to present our computer-simulation results which substantiate the efficacy and superior performance of the proposed neural network algorithm.

5 Simulative Verification

For illustrative purposes, we choose the following target function as our example: $\phi(x) = \cos^4 x/(x+1) + (x-1)/(2x+1) + e^{-x}$. Assume that we have the training data-set $\{(x_i, \phi_i), i = 1, 2, \dots, 201\}$ by sampling uniformly over interval $[0, 1]$ with step-size $x_{i+1} - x_i = 0.005$. Then, the constructed Laguerre neural network is simulated and compared. The detailed results are shown in Table 1 together with Figs. 2 and 3.

Table 1. Simulation results about the proposed Laguerre-neural-network algorithm

Target Precision	Training BPE	Optimal n	Runtime (s)	Testing BPE
10^{-4}	3.519×10^{-5}	4	0.01326	3.361×10^{-5}
10^{-5}	7.463×10^{-7}	5	0.01337	6.986×10^{-7}
10^{-6}	7.463×10^{-7}	5	0.01339	6.986×10^{-7}
10^{-7}	5.109×10^{-8}	6	0.01379	4.722×10^{-8}
10^{-8}	6.321×10^{-9}	7	0.01425	5.828×10^{-9}

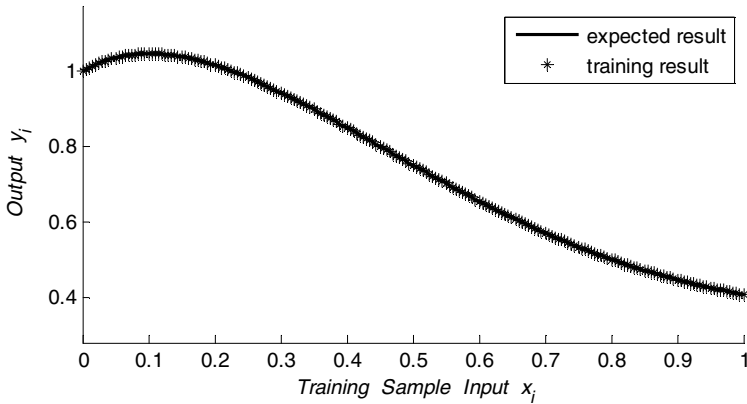


Fig. 2. Training performance of Laguerre orthogonal basis NN (under target precision 10^{-8})

From Table 1, it is evident that, in the sense of some prescribed precision, the optimal (or to say, smallest) number of hidden-layer neurons can be determined immediately for the Laguerre-orthogonal-basis neural network by using our proposed weights-direct-determination and neural-network-growing algorithm. To count the average runtime, we have conducted the simulation for 1000 times per target-precision. As can be seen from the fourth column of the table, it takes only 0.01425 seconds to obtain the optimal number of hidden-layer neurons even when the target precision is 10^{-8} . Moreover, the training BPE is only 6.321×10^{-9} , which reveals the

excellent approximation capability of the constructed Laguerre neural network. This point is shown and substantiated further in Fig. 2.

It is worth pointing out that, as for the fifth column of Table 1 about the testing BPE, we have selected totally $200 \times 5 = 1000$ untrained points (x_i, ϕ_i) to test the Laguerre neural network. The small (and tiny) testing errors show that this Laguerre-orthogonal-basis neural network possesses a very good generalization capability.

Besides, we could apply the trained Laguerre neural network for predictive purposes, e.g., over the untrained interval $[1, 1.2]$. The result is shown in Fig. 3, where the interval $[0, 1]$ corresponds to the testing part about the untrained points as mentioned in the preceding paragraph and the fifth column of Table 1. From the figure, we can see that the Laguerre neural network could have a relatively good prediction capability over the extending neighborhood of training interval.

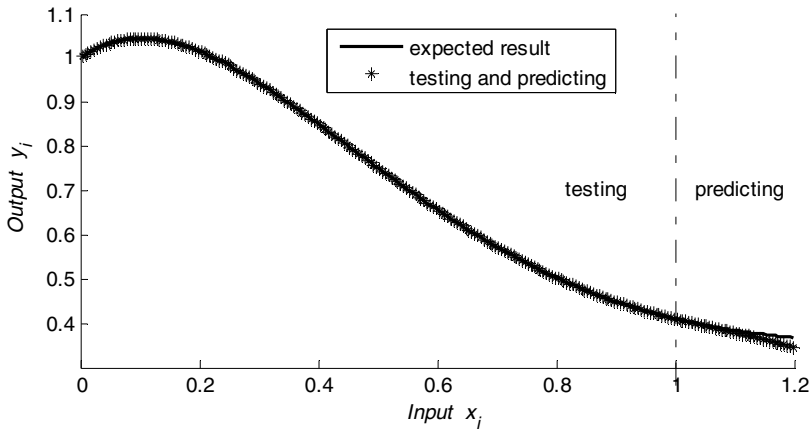


Fig. 3. Testing and predicting of Laguerre orthogonal basis NN (under target precision 10^{-8})

6 Conclusions

To remedy the weaknesses of conventional BP neural networks and their algorithms, a Laguerre-orthogonal-basis neural network has been constructed based on the polynomial curve-fitting theory in this paper. Starting from BP-type weights-updating formulas, we have derived and presented a weights-direct-determination method for calculating the optimal neural-weights directly (or to say, just in one step). Moreover, based on the weights-direct-determination method, we have also developed a neural-network growing algorithm to decide the optimal number of hidden-layer neurons efficiently and effectively. The longstanding difficulty of network-structure design might have thus been alleviated (and possibly removed) by these proposed neural-net framework. Theoretical analysis and simulation results have both substantiated the efficacy of the constructed Laguerre neural network and its growing algorithm.

Acknowledgments. This work is funded by National Science Foundation of China under Grant (60643004 and 60775050) and by the Science and Technology Office of Sun Yat-Sen University (SYSU). Before joining SYSU in 2006, Y. Zhang had been with National University of Ireland, University of Strathclyde, National University of Singapore, Chinese University of Hong Kong, South China University of Technology, and Huazhong University of Science and Technology, since 1992. Yunong had been supported by those research fellowships, assistantship and studentships. His web-page is now available at <http://www.ee.sysu.edu.cn/teacher/detail.asp?sn=129>.

References

1. Zhang, Y., Wang, J.: Recurrent Neural Networks for Nonlinear Output Regulation. *Automatica* 37(8), 1161–1173 (2001)
2. Zhang, Y., Ge, S.S., Lee, T.H.: A Unified Quadratic-Programming-Based Dynamical System Approach to Joint Torque Optimization of Physically Constrained Redundant Manipulators. *IEEE Transactions on Systems, Man, and Cybernetics* 34(5), 2126–2132 (2004)
3. Zhang, Y., Jiang, D., Wang, J.: A Recurrent Neural Network for Solving Sylvester Equation with Time-Varying Coefficients. *IEEE Transactions on Neural Networks* 13(5), 1053–1063 (2002)
4. Zhang, Y., Wang, J.: Global Exponential Stability of Recurrent Neural Networks for Synthesizing Linear Feedback Control Systems via Pole Assignment. *IEEE Transactions on Neural Networks* 13(3), 633–644 (2002)
5. Zhang, Y., Ge, S.S.: Design and Analysis of a General Recurrent Neural Network Model for Time-Varying Matrix Inversion. *IEEE Transactions on Neural Networks* 16(6), 1477–1490 (2005)
6. Lippmann, R.P.: An Introduction to Computing with Neural Nets. *IEEE ASSP Magazine* 4(2), 4–22 (1987)
7. Carl, G.L.: Advances in Feedforward Neural Networks: Demystifying Knowledge Acquiring Black Boxes. *IEEE Transactions on Knowledge and Data Engineering* 8(2), 211–226 (1996)
8. Daniel, S.Y., Zeng, X.Q.: Hidden Neuron Pruning for Multilayer Perceptrons Using a Sensitivity Measure. In: *Proceedings of the First International Conference on Machine Learning and Cybernetics*, pp. 1751–1757. IEEE Press, New York (2002)
9. Niu, B., Zhu, Y.L., Hu, K.Y., Li, S.F., He, X.X.: A Cooperative Evolutionary System for Designing Neural Networks. In: Huang, D.-S., Li, K., Irwin, G.W. (eds.) *ICIC 2006*. LNCS, vol. 4113, pp. 12–21. Springer, Heidelberg (2006)
10. Wilson, D.R., Martinez, T.R.: The Need for Small Learning Rates on Large Problems. In: *Proceedings of International Joint Conference on Neural Networks*, vol. 1, pp. 115–119. IEEE Press, New York (2001)
11. Yu, X.H., Chen, G.A.: On the Local Minima Free Condition of Backpropagation Learning. *IEEE Transactions on Neural Networks* 6(5), 1300–1303 (1995)
12. Lin, C.S.: *Numerical Analysis* (in Chinese). Science Press, Beijing (2006)
13. Kincaid, D., Cheney, W.: *Numerical Analysis: Mathematics of Scientific Computing*. China Machine Press, Beijing (2003)

MATLAB Simulation and Comparison of Zhang Neural Network and Gradient Neural Network for Online Solution of Linear Time-Varying Matrix Equation $AXB - C = 0$

Ke Chen^{1,*}, Shuai Yue^{1,*}, and Yunong Zhang^{2,*}

¹ School of Software

² School of Information Science and Technology
Sun Yat-Sen University, Guangzhou 510275, China

* jointly with the first authorship

ynzhang@ieee.org, zhnong@mail.sysu.edu.cn

Abstract. Different from gradient neural networks (GNN), a special kind of recurrent neural networks has been proposed recently by Zhang *et al* for solving online linear matrix equations with time-varying coefficients. Such recurrent neural networks, designed based on a matrix-valued error-function, could achieve global exponential convergence when solving online time-varying problems in comparison with gradient neural networks. This paper investigates the MATLAB simulation of Zhang neural networks (ZNN) for real-time solution of linear time-varying matrix equation $AXB - C = 0$. Gradient neural networks are simulated and compared as well. Simulation results substantiate the theoretical analysis and efficacy of ZNN on linear time-varying matrix equation solving.

Keywords: Linear time-varying matrix equation; Recurrent neural networks, Gradient neural networks, MATLAB simulation, Efficacy.

1 Introduction

The problem of linear matrix equations solving (including matrix-inverse problems) is considered to be a very fundamental problem widely encountered in science and engineering. It could usually be an essential part of many solutions; e.g., in control system design [1][2], signal processing [3][4], and robot inverse kinematics [5]. In view of these, we consider in this paper the following general problem formulation of linear matrix equation: $AXB - C = 0$, where coefficient matrices $A \in R^{m \times m}$, $B \in R^{n \times n}$ and $C \in R^{m \times n}$, while $X \in R^{m \times n}$ is the unknown matrix to be found. Evidently, we know that when $B = C = I$ and $m = n$, the problem reduces to the matrix-inversion problem.

As a general type of solution to this kind of matrix algebra problems, many parallel-processing computational methods have been developed, analyzed, and implemented on specific architectures [4]-[17]; especially, the ones based on recurrent neural networks (RNN) [4][8][11]-[14]. Different from conventional gradient-based neural networks for constant problems solving [2][4][8][11][16][18]-[20], a

special kind of recurrent neural networks has recently been proposed by Zhang *et al* [12]-[14] for real-time solution of time-varying matrix or vector problems. In other words, in our context of $AXB = C$, coefficient matrices A , B and C could be $A(t)$, $B(t)$ and $C(t)$, the time-varying ones. In this paper, we generalize such a design method for solving online linear time-varying matrix equation, $A(t)X(t)B(t) = C(t)$ over time $t \in [0, +\infty)$. More importantly, for the hardware/circuit implementation of recurrent neural networks, it seems necessary for us to investigate the detailed process and techniques of MATLAB simulation and verification of such resultant ZNN models. At the end of this introductory section, it is worth mentioning that the main difference between this paper and our previous work [13][14] lies in the problems and solutions being completely different (though the method is the same and named after Zhang *et al*).

2 Theoretical Analysis

In the ensuing subsections, ZNN model is designed and exploited to solve online the linear time-varying matrix equation, $A(t)X(t)B(t) - C(t) = 0$, in real time t , of which the solution is compared with conventional GNN models'.

2.1 Problem Formulation

Consider the following linear time-varying matrix equation

$$A(t)X(t)B(t) - C(t) = 0, \quad \forall t \in [0, +\infty) \quad (1)$$

where $A(t) \in R^{m \times m}$, $B(t) \in R^{n \times n}$ and $C(t) \in R^{m \times n}$ are smoothly time-varying coefficient matrices, and $X(t) \in R^{m \times n}$ is the unknown matrix to be obtained. It is worth mentioning here that the time derivatives of matrices $A(t)$, $B(t)$ and $C(t)$ are assumed to be known or estimated readily as $\dot{A}(t)$, $\dot{B}(t)$ and $\dot{C}(t)$.

It follows from Kronecker-product and vectorization techniques [14][21] that linear time-varying matrix equation (1) is equivalent to

$$(B^T(t) \otimes A(t))\text{vec}(X(t)) = \text{vec}(C(t)). \quad (2)$$

Symbol \otimes denotes the Kronecker product; i.e., $P \otimes Q$ is a larger matrix made by replacing the ij th entry p_{ij} of P with matrix $p_{ij}Q$. Operator $\text{vec}(X) \in R^{mn}$ generates a column vector obtained by stacking all column vectors of X together. Thus, before solving the linear time-varying matrix equation (1), the following unique-solution condition could be derived to lay a basis for further discussion.

Unique-solution condition: Linear matrix equation (1) is uniquely solvable, if all eigenvalues of matrices $A(t) \in R^{m \times m}$ and $B(t) \in R^{n \times n}$ are nonzero at any time instant $t \in [0, +\infty)$. \square

2.2 ZNN and GNN Solvers

By following the design method of Zhang *et al* [12]-[14], we can derive the following implicit dynamics of ZNN model which solves linear time-varying matrix equation (1) in real time t :

$$A(t)\dot{X}(t)B(t) = -\dot{A}(t)X(t)B(t) - A(t)X(t)\dot{B}(t) + \dot{C}(t) - \gamma\mathcal{F}(A(t)X(t)B(t) - C(t)), \quad (3)$$

where $X(t)$, starting from an initial condition $X(0) := X_0 \in R^{m \times n}$, is the activated neural-state matrix corresponding to the theoretical solution $X^*(t)$ of (1). In addition, $\gamma > 0 \in R$, $E(t) := A(t)X(t)B(t) - C(t) \in R^{m \times n}$, and each element of activation function array $\mathcal{F}(\cdot)$ could take the following form of power-sigmoid activation function (with $\xi \geq 2$ and $p \geq 3$):

$$f(e_{ij}) = \begin{cases} e_{ij}^p, & \text{if } |e_{ij}| \geq 1, \\ \frac{1+\exp(-\xi)}{1-\exp(-\xi)} \cdot \frac{1-\exp(-\xi e_{ij})}{1+\exp(-\xi e_{ij})}, & \text{otherwise.} \end{cases} \quad (4)$$

Moreover, when using linear activation-function array $\mathcal{F}(E) = E$, ZNN model (3) reduces to the following linear one:

$$A\dot{X}B = -\dot{A}XB - AX\dot{B} - \gamma AXB + (\dot{C} + \gamma C). \quad (5)$$

For comparison, to solve the linear time-varying matrix equation (1) but with constant coefficient matrices A , B and C , we could have a linear gradient neural network (GNN), $\dot{X} = -\gamma A^T (AXB - C) B^T$ by using the negative-gradient method, and then have a generalized nonlinear GNN model,

$$\dot{X}(t) = -\gamma A^T(t)\mathcal{F}(A(t)X(t)B(t) - C(t))B^T(t), \quad (6)$$

of which the convergence could be proved only for the situation with constant coefficient matrices A , B and C [2][11][19].

2.3 ZNN Convergence

While the above subsections present a general description of ZNN models (3) and (5), detailed design consideration and main theoretical results about their global exponential convergence are given in this subsection. The following propositions might be important.

Proposition 1. *Consider smoothly time-varying coefficient matrices $A(t) \in R^{m \times m}$, $B(t) \in R^{n \times n}$ and $C(t) \in R^{m \times n}$ of linear matrix equation (1), with the unique-solution condition satisfied. If a monotonically-increasing odd activation function processing-array $\mathcal{F}(\cdot)$ is used, then state matrix $X(t)$ of ZNN model (3), starting from any initial state $X_0 \in R^{m \times n}$, converges to the theoretical solution $X^*(t)$ of linear time-varying matrix equation (1). \square*

Proposition 2. *In addition to Proposition 2, ZNN (3) possesses the following properties. If linear activation function $f(e_{ij}) = e_{ij}$ is used, then global exponential convergence [in terms of error $E(t)$] could be achieved for (3) with rate γ . If power-sigmoid function (4) is used, then superior convergence for error range $(-\infty, +\infty)$ could be achieved for (3), as compared to the linear case. \square*

3 Simulation Techniques

In this section, we investigate the MATLAB simulation techniques for ZNN (3).

3.1 Kronecker Product and Vectorization

For the proposes of MATLAB simulation, we have to transform the matrix-form differential equation (3) [including (5) as a special case] into a vector-form differential equation.

Proposition 3. *The matrix-form ZNN model (3) can be transformed to the vector-form differential equation:*

$$M(t)\dot{x}(t) = -\dot{M}(t)x(t) + \text{vec}(\dot{C}(t)) - \gamma\mathcal{F}(M(t)x(t) - \text{vec}(C(t))), \quad (7)$$

where mass matrix $M(t) := B^T(t) \otimes A(t)$, integration vector $x(t) := \text{vec}(X(t))$, and activation-function mapping $\mathcal{F}(\cdot)$ is defined the same as in (3) except that its dimensions are flexibly changed hereafter as $\mathcal{F}(\cdot) : R^{mn \times 1} \rightarrow R^{mn \times 1}$. \square

To generate the mass matrix $M = B^T \otimes A$ from matrices A and B , we could simply use MATLAB routine “kron” as follows (to be an example):

```
function output=MatrixM(t,x)
A=[sin(t) cos(t); -cos(t) sin(t)];
B=[2 cos(t)+sin(t) sin(t);-cos(t) 2 cos(t);sin(t)-2 2-sin(t) 2-sin(t)];
output=kron(B',A);
```

Based on MATLAB routine “reshape”, the vectorization of a matrix could be achieved readily as well. For example, the following MATLAB code is used to evaluate the right-hand side of differential equation (7) [equivalently, (3)], where “DiffA”, “DiffB”, “DiffC” and “DiffM” denote the MATLAB functions which evaluate the time derivatives of coefficients $A(t)$, $B(t)$, $C(t)$ and mass $M(t)$.

```
function y=ZNNRightHandSide(t,x,gamma)
if nargin==2, gamma=1; end
A=LTVEmatrixA(t,x); B=LTVEmatrixB(t,x); C=LTVEmatrixC(t,x);
dotA=DiffA(t,x); dotB=DiffB(t,x); dotC=DiffC(t,x);
[m,n]=size(C); vecC=reshape(C,m*n,1); vecDotC=reshape(dotC,m*n,1);
M1=kron(B',dotA); M2=kron(dotB',A); M3=kron(B',A);
y=-M1*x-M2*x+vecDotC-gamma*AFMpowersigmoid(M3*x-vecC);
```

Similar to Proposition 3, the matrix-form gradient neural network (6) can be transformed to the following vector-form differential equation:

$$\dot{x}(t) = -\gamma (B(t) \otimes A^T(t)) \mathcal{F}((B^T(t) \otimes A(t)) \text{vec}(X(t)) - \text{vec}(C(t))). \quad (8)$$

The MATLAB code for simulating such a gradient neural network is presented below for comparison. It returns the evaluation of the right-hand side of GNN model (8) [equivalently, (6)].

```
function y=GNNRightHandSide(t,x,gamma)
if nargin==2, gamma=1; end
A=LTVEmatrixA(t,x); B=LTVEmatrixB(t,x); C=LTVEmatrixC(t,x);
[m,n]=size(C); vecC=reshape(C,m*n,1);
M3=kron(B,A'); M4=kron(B',A);
y=-gamma*M3*AFMpowersigmoid(M4*x-vecC);
```

3.2 ODE with Mass Matrix

For the simulation of ZNN model (3) [equivalently, (7)], the MATLAB routine “ode45” is preferred. This is because “ode45” can solve initial-value ordinary differential equation (ODE) problems with nonsingular mass matrices, e.g., $M(t, x)\dot{x} = g(t, x)$, $x(0) = x_0$, where matrix $M(t, x)$ on the left-hand side of such an equation is termed the mass matrix, and the right-hand side $g(t, x) := -\dot{M}x + \text{vec}(\dot{C}) - \gamma\mathcal{F}(Mx - \text{vec}(C))$ for our case.

To solve an ODE problem with a mass matrix, the MATLAB routine “odeset” should also be used. Its “Mass” property should be assigned to be the function handle “@MatrixM”, which returns the evaluation of mass matrix $M(t, x)$. Note that, if 1) $M(t, x)$ does not depend on state variable x and 2) the function “MatrixM” is to be invoked with only one input argument t , then the “MStateDep” property of “odeset” should be “none”. For example, the following MATLAB code can be used to solve an initial-value ODE problem with state-independent mass matrix $M(t)$ and starting from a random initial state x_0 .

```
tspan=[0 10];
x0=4*(rand(6,1)-0.5*ones(6,1));
options=odeset('Mass',@MatrixM,'MStateDep','none');
[t,x]=ode45(@ZNNRightHandSide,tspan,x0,options,gamma);
```

4 Illustrative Examples

For simulation and comparison purposes, let us consider linear matrix equation (1) with the following time-varying coefficient matrices $A(t)$, $B(t)$ and $C(t)$:

$$A(t) = \begin{bmatrix} \sin t & \cos t \\ -\cos t & \sin t \end{bmatrix}, \quad B(t) = \begin{bmatrix} 2 & \sin t + \cos t & \sin t \\ -\cos t & 2 & \cos t \\ \sin t - 2 & 2 - \sin t & 2 - \sin t \end{bmatrix},$$

and $C(t) = [2 + \sin t \cos t - 2 \cos t, 3 \cos t + \sin t - \sin t \cos t, \sin t + 2 \cos t - \sin t \cos t; -\cos t + \sin^2 t - 2 \sin t, 2 + 2 \sin t - \sin^2 t, \cos t + 2 \sin t - \sin^2 t]$. For checking the correctness of the neural solutions, the unique time-varying theoretical solution $X^*(t)$ could be given for this specific problem as the following:

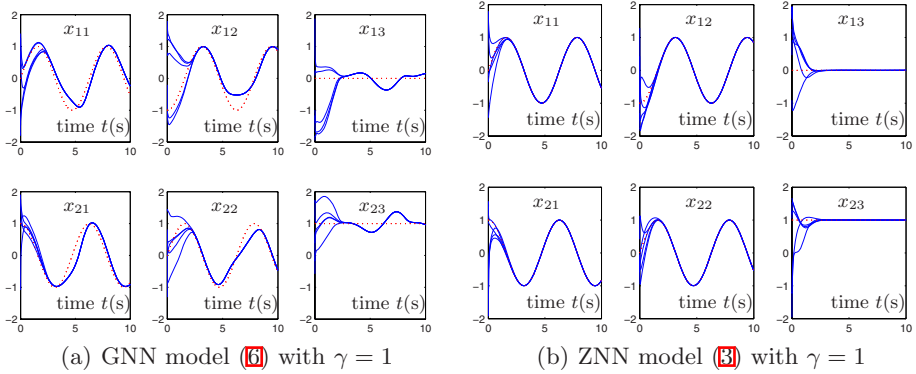


Fig. 1. Online solution of linear time-varying matrix equation (11) by GNN model (6) and ZNN model (3) with $\gamma = 1$ and using a power-sigmoid function array

$$X^*(t) = \begin{bmatrix} \sin t & -\cos t & 0 \\ \cos t & \sin t & 1 \end{bmatrix}.$$

To solve linear matrix equation (11) with the above coefficients, both GNN model (6) and ZNN model (3) are employed with the following MATLAB code.

```
function GNNConvergence(gamma)
for iter=1:5
x0=4*(rand(6,1)-0.5*ones(6,1)); tspan=[0 10];
[t,x]=ode45(@GNNRightHandSide,tspan,x0,gamma);
xStar=[sin(t) cos(t) -cos(t) sin(t) ...
zeros(length(t),1) ones(length(t),1)];
for k=1:6
j=[1 4 2 5 3 6]; subplot(2,3,j(k));
plot(t,x(:,k)); hold on; plot(t,xStar(:,k),'r:'); hold on; end
end
```

```
function ZNNConvergence(gamma)
tspan=[0 10]; options=odeset('Mass',@MatrixM,'MStateDep','none');
for iter=1:5
x0=4*(rand(6,1)-0.5*ones(6,1));
[t,x]=ode45(@ZNNRightHandSide,tspan,x0,options,gamma);
xStar=[sin(t) cos(t) -cos(t) sin(t) ...
zeros(length(t),1) ones(length(t),1)];
for k=1:6
j=[1 4 2 5 3 6]; subplot(2,3,j(k));
plot(t,x(:,k)); hold on;
plot(t,xStar(:,k),'r:'); hold on; end
end
```

By using the above user-defined MATLAB functions “GNNConvergence” and “ZNNConvergence” with input arguments $\gamma = 1$, we can generate Fig. 1(a) and (b). As seen from the figure, superior convergence has been achieved by

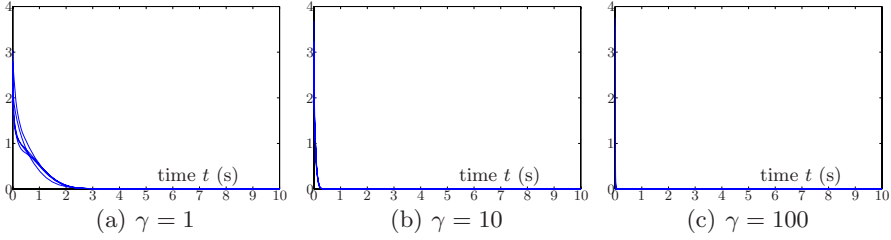


Fig. 2. Convergence of computational error $\|X(t) - X^*(t)\|$ synthesized by ZNN (3) with different values of design parameter γ and using a power-sigmoid processing array

ZNN model (3), as compared to GNN model (6) (which, at the present form, seems unable to solve exactly this time-varying problem). Moreover, design parameter γ has remarkable effectiveness on the ZNN convergence rate. By calling “ZNNConvergence” with $\gamma = 10$ and 100, we can generate two more figures and observe that the larger the design parameter γ is, the faster the recurrent neural network converges. This point could also be shown by monitoring the computational error $\|X(t) - X^*(t)\|$, i.e., Fig. 2.

In addition, it is worth mentioning that using power-sigmoid activation functions has a smaller steady-state residual error than using linear activation functions. Compared to the case of using linear activation functions, superior performance could also be achieved by using power-sigmoid functions under the same design parameters and conditions.

5 Conclusions

By following the design method of Zhang *et al.*, we have developed the ZNN models for solve online the linear time-varying matrix equation $A(t)X(t)B(t) - C(t) = 0$. By using power-sigmoid activation functions, such ZNN models have been investigated and simulated. Several important MATLAB simulation techniques have been introduced, such as, Kronecker product of matrices and MATLAB routine “ode45” with mass-matrix property. Computer-simulation results have demonstrated the effectiveness and efficiency of ZNN models for solving online linear time-varying problems.

References

1. Zhang, Y., Wang, J.: Recurrent Neural Networks for Nonlinear Output Regulation. *Automatica* 37, 1161–1173 (2001)
2. Zhang, Y., Wang, J.: Global Exponential Stability of Recurrent Neural Networks for Synthesizing Linear Feedback Control Systems via Pole Assignment. *IEEE Transactions on Neural Networks* 13, 633–644 (2002)
3. Zhang, Y., Leithead, W.E., Leith, D.J.: Time-Series Gaussian Process Regression Based on Toeplitz Computation of $O(N^2)$ Operations and $O(N)$ -level Storage. In: *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 3711–3716 (2005)

4. Steriti, R.J., Fiddy, M.A.: Regularized Image Reconstruction Using SVD and a Neural Network Method for Matrix Inversion. *IEEE Transactions on Signal Processing* 41, 3074–3077 (1993)
5. Sturges Jr., R.H.: Analog Matrix Inversion (Robot Kinematics). *IEEE Journal of Robotics and Automation* 4, 157–162 (1988)
6. Carneiro, N.C.F., Caloba, L.P.: A New Algorithm for Analog Matrix Inversion. *Proceedings of the 38th Midwest Symposium on Circuits and Systems* 1, 401–404 (1995)
7. El-Amawy, A.: A Systolic Architecture for Fast Dense Matrix Inversion. *IEEE Transactions on Computers* 38, 449–455 (1989)
8. Jang, J., Lee, S., Shin, S.: An Optimization Network for Matrix Inversion. In: *Neural Information Processing Systems*. American Institute of Physics, pp. 397–401 (1988)
9. Wang, Y.Q., Gooi, H.B.: New Ordering Methods for Space Matrix Inversion via Diagonalization. *IEEE Transactions on Power Systems* 12, 1298–1305 (1997)
10. Yeung, K.S., Kumbi, F.: Symbolic Matrix Inversion with Application to Electronic Circuits. *IEEE Transactions on Circuits and Systems* 35, 235–238 (1988)
11. Zhang, Y.: Revisit the Analog Computer and Gradient-Based Neural System for Matrix Inversion. In: *Proceedings of IEEE International Symposium on Intelligent Control*, pp. 1411–1416 (2005)
12. Zhang, Y., Ge, S.S.: A General Recurrent Neural Network Model for Time-Varying Matrix Inversion. In: *Proceedings of the 42nd IEEE Conference on Decision and Control*, pp. 6169–6174 (2003)
13. Zhang, Y., Ge, S.S.: Design and Analysis of a General Recurrent Neural Network Model for Time-Varying Matrix Inversion. *IEEE Transactions on Neural Networks* 16, 1477–1490 (2005)
14. Zhang, Y., Jiang, D., Wang, J.: A Recurrent Neural Network for Solving Sylvester Equation with Time-Varying Coefficients. *IEEE Transactions on Neural Networks* 13, 1053–1063 (2002)
15. Mead, C.: *Analog VLSI and Neural Systems*. Addison-Wesley, Reading (1989)
16. Zhang, Y., Chen, K., Ma, W., Li, X.: Matlab Simulation of Gradient-based Neural Network for Online Matrix Inversion. In: Huang, D.-S., Heutte, L., Loog, M. (eds.) *ICIC 2007*. LNCS (LNAI), vol. 4682, pp. 98–109. Springer, Heidelberg (2007)
17. Zhang, Y., Chen, K., Ma, W.: Matlab Simulation and Comparison of Zhang Neural Network and Gradient Neural Network of Online Solution of Linear Time-Varying Equations. In: *DCDIS Proceedings of International Conference on Life System Modeling and Simulation*, pp. 450–454 (2007)
18. Manherz, R.K., Jordan, B.W., Hakimi, S.L.: Analog Methods for Computation of the Generalized Inverse. *IEEE Transactions on Automatic Control* 13, 582–585 (1968)
19. Zhang, Y.: A Set of Nonlinear Equations and Inequalities Arising in Robotics and its Online Solution via a Primal Neural Network. *Neurocomputing* 70, 513–524 (2006)
20. Zhang, Y.: Towards Piecewise-Linear Primal Neural Networks for Optimization and Redundant Robotics. In: *Proceedings of IEEE International Conference on Networking, Sensing and Control*, pp. 374–379 (2006)
21. Horn, R.A., Johnson, C.R.: *Topics in Matrix Analysis*, pp. 239–297. Cambridge University Press, Cambridge (1991)
22. Anderson, J.A., Rosenfeld, E.: *Neurocomputing: Foundations of Research*. The MIT Press, Cambridge (1988)

Motion Detection Using Spiking Neural Network Model

QingXiang Wu^{1,2}, T.M. McGinnity¹, Liam Maguire¹, Jianyong Cai²,
and G.D. Valderrama-Gonzalez¹

¹ Intelligent Systems Research Centre, University of Ulster at Magee Campus
Derry, BT48 7JL, Northern Ireland, UK

{q.wu, tm.mcginny, lp.Maguire, g.valderrama}@ulster.ac.uk

² School of School of Physics and OptoElectronic Technology, Fujian Normal University
Fuzhou, 350007, China

{qxwu, c jy}@fjnu.edu.cn

Abstract. Inspired by the behaviour of the human visual system, a spiking neural network is proposed to detect moving objects in a visual image sequence. The structure and the properties of the network are detailed in this paper. Simulation results show that the network is able to perform motion detection for dynamic visual image sequence. Boundaries of moving objects are extracted from an active neuron group. Using the boundary, a moving object filter is created to take the moving objects from the grey image. The moving object images can be used to recognise moving objects. The moving tracks can be recorded for further analysis of behaviours of moving objects. It is promising to apply this approach to video processing domain and robotic visual systems.

Keywords: Motion detection; spiking neural networks; visual system.

1 Introduction

A football player can promptly perform a series of actions to capture a football when he sees the moving football toward him. The information of the moving football conveys to the brain through the visual system. The retina contains complex circuits of neurons that extract salient information from visual inputs. Signals from photoreceptors are processed by retinal interneurons, integrated by retinal ganglion cells and sent to the brain by axons of retinal ganglion cells. Different cells respond to different visual features, such as light intensity, colour or moving objects [1–5]. Mammalian retinas contain approximately 55 distinct cell types, each with a different function [1]. A retinal cell type responds to upward motion has been identified in [6]. Results in [7] demonstrate that information for segmenting scenes by relative motion is represented as early as visual cortex V1. To detect moving objects, the brain must distinguish local motion within the scene from the global image. The findings in [8] show how a population of ganglion cells selective for differential motion can rapidly flag moving objects, and even segregate multiple moving objects. In [9], it is shown that neurons compute internal models of the physical laws of motion. These findings are shown some principles for the brain to detect moving objects in the psychological or

statistical level. What are the exact neuronal circuits for motion detection? How can we simulate the neuronal circuits in electronic circuits and then apply them to artificial intelligent systems? This is the motivation of this paper. Jeffress [22-24] applied the time difference principle of axonal delay to account for sound localisation [11,13]. Based on spiking neuron model and axonal delay [10-14], a neuronal circuit is proposed to explain how a spiking neural network can detect moving objects in an image sequence. The neuronal circuit has been simulated in software and embedded in a simulation system. Combining with the traditional image processing approaches, the system can demonstrate retrieval of moving objects from an image sequence.

The remainder of this paper is organized as follows. In Section 2, axonal delays are used to construct a spiking neural network which is used to simulate the visual cortex for motion detection, and the principle of motion detection is described. The network model is based on conductance-based integrate-and-fire neurons. The behaviours of the neural network with the axonal delay are represented by a set of equations in Section 3. Simulation system and results for motion detection are presented in Section 4. Discussions about the network are given in Section 5.

2 Spiking Neural Network Model for Motion Detection

The human visual system performs motion detection very efficiently. Neuroscientists have found that there are various receptive fields from simple cells in the striate cortex to those of the retina and lateral geniculate nucleus (see page 236-248 in [15]), and the axonal delay causes a phase shift for a spike train [10-14]. Inspired by the axonal delay mechanism, a spiking neural network model is proposed to detect moving objects. Its structure is shown in Fig. 1. Suppose that the first layer represents photonic receptors for an image from visual system. Each pixel of the image corresponds to a receptor. The intermediate layer is composed of two neuron arrays. N1 neuron array and N2 neuron array have the same size as the receptor layer. N1 and N2 neuron array are connected to neurons in output layer. As shown in Fig.1, receptor $N_r(x, y)$ is connected to $N1(x, y)$ through excitatory synapse without delay and through inhibitory synapse with an axonal delay Δt . Similarly, the neuron $N_r(x, y)$ is also connected to $N2(x, y)$ through excitatory synapse with an axonal delay Δt and through inhibitory synapse without delay. Let $S_{N_r}(x, y, t)$ represent current from receptor $N_r(x, y)$. If the current from receptor $N_r(x, y)$ is stable, i.e. current $S_{N_r}(x, y, t)$ is equal to current $S_{N_r}(x, y, t-\Delta t)$, the excitatory input and the inhibitory input of neuron $N1(x, y)$ can be balanced by adjusting the parameters of synapses, and then neuron $N1(x, y)$ is silent. If the current of receptor $N_r(x, y)$ becomes stronger, i.e. the current $S_{N_r}(x, y, t)$ is larger than current $S_{N_r}(x, y, t-\Delta t)$, the balance is broken, and then neuron $N1(x, y)$ will generate spikes if $S_{N_r}(x, y, t)$ is larger enough than $S_{N_r}(x, y, t-\Delta t)$. If the current of receptor $N_r(x, y)$ becomes weaker, neuron $N1(x, y)$ does not fire. In this case, input neuron $N2(x, y)$ through inhibitory synapse becomes weaker, but input through excitatory synapse is still strong. Neuron $N2(x, y)$ will fire if $S_{N_r}(x, y, t)$ is smaller enough than $S_{N_r}(x, y, t-\Delta t)$. Therefore, the gray scale changes of pixels in the image are reflected in the output neuron layer, i.e. Neuron $N(x', y')$ will fire if Neuron $N1$ or Neuron $N2$ fires. Therefore, the moving object corresponds to high firing-rate

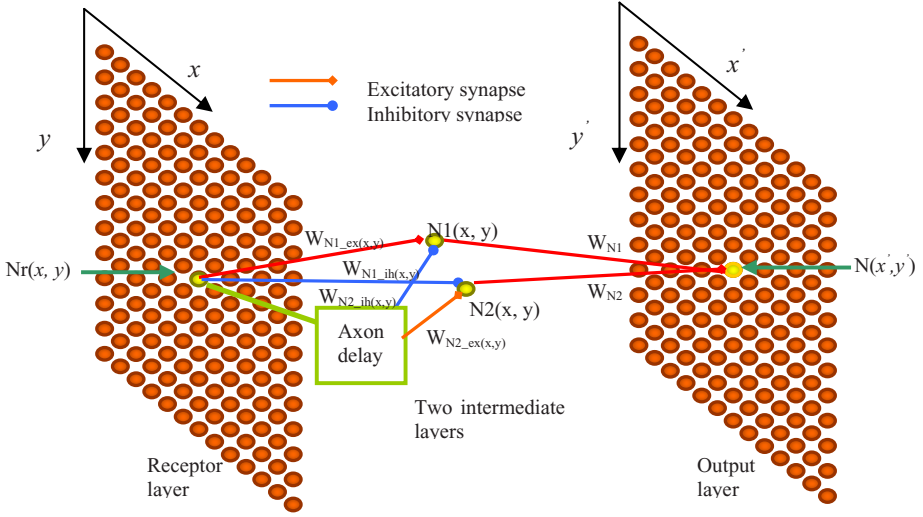


Fig. 1. Spiking neural network model for motion detection

neurons in the output layer. The object can be obtained by binding these highly active neurons in the output layer.

3 Spiking Neuron Model and Simulation Algorithms

Simulation results show that the conductance-based integrate-and-fire model is very close to the Hodgkin and Huxley neuron model [16-21]. Therefore, this model is applied to the aforementioned network model. Let $G_{x,y}(t)$ represent gray scale of image pixel at point (x,y) at time t , $q_{x,y}^{ex}(t)$ represent peak conductance caused by excitatory current $S_{Nr}(x, y, t)$ from a receptor at point (x,y) , and $q_{x,y}^{ih}(t)$ represent peak conductance caused to inhibitory current $S_{Nr}(x, y, t)$ from a receptor at point (x,y) . For simplicity, suppose that each receptor can transform a gray scale value to peak conductance by the following expressions.

$$q_{x,y}^{ex}(t) = \alpha G_{x,y}(t); \quad q_{x,y}^{ih}(t) = \beta G_{x,y}(t) \quad (1)$$

where α and β are constants. According to the conductance based integrate-and-fire model [20-21], neuron N1 is governed by the following equations.

$$\frac{dg_{N1_ex(x,y)}(t)}{dt} = -\frac{1}{\tau_{ex}} g_{N1_ex(x,y)}(t) + \alpha G_{x,y}(t). \quad (2)$$

$$\frac{dg_{N1_ih(x,y)}(t)}{dt} = -\frac{1}{\tau_{ih}} g_{N1_ih(x,y)}(t) + \beta G_{x,y}(t). \quad (3)$$

$$c_m \frac{dv_{N1(x,y)}(t)}{dt} = g_l(E_l - v_{N1(x,y)}(t)) + \frac{w_{N1_ex(x,y)}g_{N1_ex(x,y)}(t)}{A_{ex}}(E_{ex} - v_{N1(x,y)}(t)) + \frac{w_{N1_ih(x,y)}g_{N1_ih(x,y)}(t-\Delta t)}{A_{ih}}(E_{ih} - v_{N1(x,y)}(t)), \quad (4)$$

where $g_{N1_ex(x,y)}(t)$ and $g_{N1_ih(x,y)}(t)$ are the conductance for excitatory and inhibitory synapses respectively, τ_{ex} and τ_{ih} are the time constants for excitatory and inhibitory synapses respectively, Δt is the axonal delay, $v_{N1(x,y)}(t)$ is the membrane potential of neuron $N1$, E_{ex} and E_{ih} are the reverse potential for excitatory and inhibitory synapses respectively, c_m represents a capacitance of the membrane, g_l represents the conductance of membrane, ex is short for excitatory and ih for inhibitory, A_{ex} is the membrane surface area connected to a excitatory synapse, and A_{ih} is the membrane surface area connected to a inhibitory synapse, $w_{N1_ih(x,y)}$ represents the strength of inhibitory synapses, $w_{N1_ex(x,y)}$ represents the strength of excitatory synapses. $w_{N1_ih(x,y)}$ and $w_{N1_ex(x,y)}$ are adjusted so that neuron $N1$ dose not fire when $G_{x,y}(t) = G_{x,y}(t - \Delta t)$. By analogy, membrane potential of Neuron $N2$ is governed by the equation as follows.

$$c_m \frac{dv_{N2(x,y)}(t)}{dt} = g_l(E_l - v_{N2(x,y)}(t)) + \frac{w_{N2_ex(x,y)}g_{N2_ex(x,y)}(t-\Delta t)}{A_{ex}}(E_{ex} - v_{N2(x,y)}(t)) + \frac{w_{N2_ih(x,y)}g_{N2_ih(x,y)}(t)}{A_{ih}}(E_{ih} - v_{N2(x,y)}(t)), \quad (5)$$

where $v_{N2(x,y)}(t)$ represents the membrane potential of neuron $N2$. Note that changes of conductance of excitatory synapses have a delay comparing with Neuron $N1$, but changes of conductance of inhibitory synapses have not any delay that is different from Neuron $N1$. When the membrane potential of Neuron $N1$ and $N2$ reaches a threshold v_{th} the neuron generates a spike respectively. These spikes are transferred to corresponding neuron in output layer. Let $S_{N1}(t)$ represent a spike train which is generated by neuron $N1$.

$$S_{N1}(t) = \begin{cases} 1 & \text{if neuron } N1 \text{ fires at } timet. \\ 0 & \text{if neuron } N1 \text{ does not fire at } timet. \end{cases} \quad (6)$$

By analogy, let $S_{N2}(t)$ represent spike trains for neurons $N2$. Neuron $N_{x',y'}$ in the output layer is governed by the following equations.

$$\frac{g_{ex(x',y')}(t)}{dt} = -\frac{1}{\tau_{ex}}g_{ex(x',y')}(t) + (w_{N1}S_{N1}(t) + w_{N2}S_{N2}(t)). \quad (7)$$

$$c_m \frac{dv_{x',y'}(t)}{dt} = g_l(E_l - v_{x',y'}(t)) + \frac{g_{ex(x',y')}(t)}{A_{ex}}(E_{ex} - v_{x',y'}(t)). \quad (8)$$

Note that intermediate neurons are connected to output Neuron $N_{x',y'}$ only by excitatory synapses. Let $S_{x',y'}(t)$ represent spike train generated by Neuron $N_{x',y'}$ in output layer. The firing rate for Neuron $N_{x',y'}$ is calculated by the following expression.

$$r_{x',y'}(t) = \frac{1}{T} \sum_t^{t+T} S_{x',y'}(t) \quad (9)$$

Plotting $r_{x',y'}(t)$ as a grey image, white areas indicate neuron groups with high firing rate. Drawing the outside boundaries of firing neuron groups, boundaries of moving objects are extracted.

4 Simulation Results

This model is used to simulate biologically inspired neuronal behaviours. Learning mechanism is not required. The parameters are set corresponding to biologic neurons. Following parameters for the network were used in the experiments. $v_{th} = -60$ mv. $v_{reset} = -70$ mv. $E_{ex} = 0$ mv. $E_{ih} = -75$ mv. $E_l = -70$ mv. $g_l = 1.0 \mu\text{s}/\text{mm}^2$. $c_m = 10$ nF/mm². $\tau_{ex} = 2$ ms. $\tau_{ih} = 2$ ms. $A_{ih} = 0.028953$ mm². $A_{ex} = 0.014103$ mm². These parameters can be slightly adjusted to get good quality of output image. The architecture of simulation system is shown in Fig. 2.

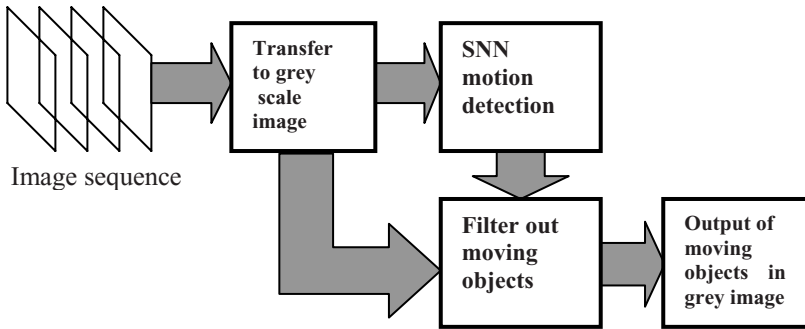


Fig. 2. The architecture of simulation system

The system takes an image from the image sequence each time step. The image is transferred to a grey scale image. The grey image presents to the spiking neural network (SNN) for motion detection. The moving objects can be detected by the SNN based on the equations in Section 3. The edges of firing neuron groups are used to determine the boundaries of the moving objects. Using the boundaries of the objects, a filter is generated to take out of moving objects from background. Therefore, the moving object in the grey image is transferred to the output image. The results of simulations are shown in Fig. 3. Images (a), (c), (e) and (g) are original image from

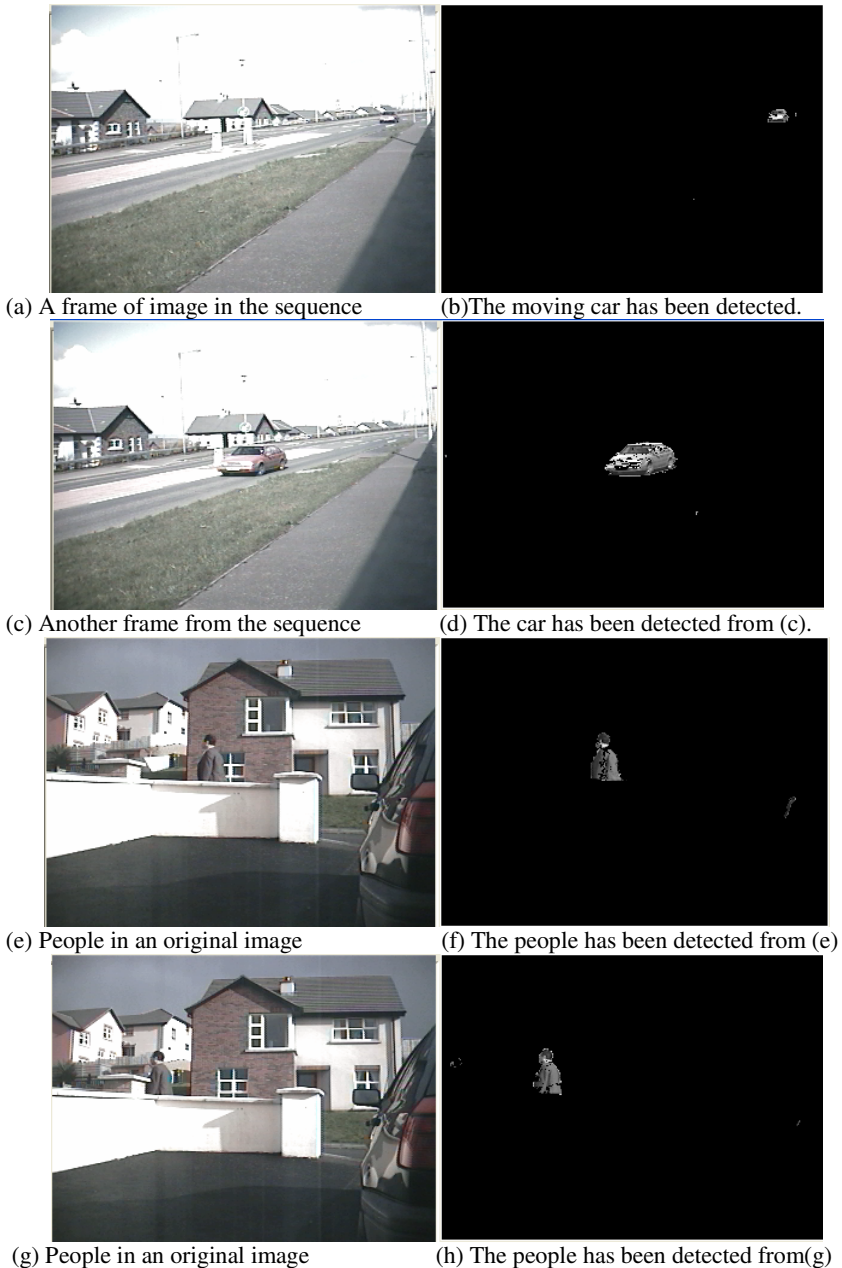


Fig. 3. Results of simulations

image sequence, where as (b), (d), (f) and (h) are corresponding outputs of the simulation system. A simplified model based on the principle is implemented using C++ in Windows XP. This program can be used to demonstrate the dynamic properties for SNN motion detection in real time.

5 Discussion

Inspired by biological findings, a neuronal circuit for motion detection is proposed in this paper. The neuronal circuit is based on axonal delay using spiking neuron model and it can be used to explain how a spiking neural network in the visual system can detect moving objects. Further research is required to establish the actual mechanisms employed by the visual cortex to determine motion. However, the proposal presented here can be used in artificial intelligent systems. Since the circuit is based on spiking neuron model, other findings in the human visual system can be integrated into the system to process more complicated moving objects tracking and recognition. It would be very promising to create more powerful image processing system using more biological principles found in the visual system, for example, this can be extended to deal with the coloured images. This is a very interesting topic for further study.

References

1. Masland, R.H.: The Fundamental Plan of the Retina. *Nature Neurosci.* 4, 877–886 (2001)
2. Wassle, H.: Parallel Processing in the Mammalian Retina. *Nature Rev. Neurosci.* 5, 747–757 (2004)
3. Nelson, R., Kolb, H.: On and off Pathways in the Vertebrate Retina and Visual System. In: Chalupa, L.M., Werner, J.S. (eds.) *The Visual Neurosciences*, pp. 260–278. MIT Press, Cambridge (2003)
4. Demb, J.B.: Cellular Mechanisms for Direction Selectivity in the Retina. *Neuron* 55, 179–186 (2007)
5. Taylor, W.R., Vaney, D.I.: New Directions in Retinal Research. *Trends Neurosci.* 26, 379–385 (2003)
6. Kim, I.J., Zhang, Y., Yamagata, M., Meister, M., Sanes, J.R.: Molecular Identification of a Retinal Cell Type that Responds to Upward Motion. *Nature* 452, 478–482 (2008)
7. Reppas, J.B., Niyogi, S., Dale, A.M., Sereno, M.I., Tootell, B.H.: Representation of Motion Boundaries in Retinotopic Human Visual Cortical Areas. *Nature* 388(6638), 175–186 (1997)
8. Olveczky, B.P., Baccus, S.A., Meister, M.: Segregation of Object and Background Motion in the Retina. *Nature* 423(6938), 401–408 (2003)
9. Angelaki, D.E., Shaikh, A.G., Green, A.M., Dickman, J.D.: Neurons Compute Internal Models of the Physical Laws of Motion. *Nature* 430(6999), 560–565 (2004)
10. Lin, J.W., Faber, D.S.: Modulation of Synaptic Delay during Synaptic Plasticity. *Trends Neurosci.* 25(9), 44–55 (2002)
11. Pena, J.L., Kazuo, S., VSaberi, F.K., Konishi, M.: Cochlear and Neural Delays for Coincidence Detection in Owls. *The Journal of Neuroscience* 21(23), 9455–9459 (2001)
12. Senn, W., Schneider, M., Ruf, B.: Activity-Dependent Development of Axonal and Dendritic Delays, or, Why Synaptic Transmission Should Be Unreliable. *Neural Computation* 14, 583–619 (2002)
13. Carr, C.E., Konishi, M.: Axonal Delay Lines for Time Measurement in the Owl's Brainstem. *Proceedings of the National Academy of Sciences of the United States of America* 85(21), 8311–8315 (1988)
14. Crook, S.M., Ermentrout, G.B., Vanier, M.C., Bower, J.M.: The Role of Axonal Delay in the Synchronization of Networks of Coupled Cortical Oscillators. *Journal of Computational Neuroscience* 4(2), 157–6873 (1997)

15. Kandel, E.R., Shwartz, J.H.: Principles of Neural Science. Edward Arnold (Publishers) Ltd. (1981)
16. Koch, C.: Biophysics of Computation: Information Processing in Single Neurons. Oxford University Press, Oxford (1999)
17. Dayan, P., Abbott, L.F.: Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems. The MIT Press, Cambridge (2001)
18. Gerstner, W., Kistler, W.: Spiking Neuron Models: Single Neurons, populations, Plasticity. Cambridge University Press, Cambridge (2002)
19. Müller, E.: Simulation of High-Conductance States in Cortical Neural Networks, Masters Thesis, University of Heidelberg, HD-KIP-03-22 (2003)
20. Wu, Q.X., McGinnity, T.M., Maguire, L.P., Glackin, B., Belatreche, A.: Learning Mechanism in Networks of Spiking Neurons. Studies in Computational Intelligence, vol. 35, pp. 171–197. Springer, Heidelberg (2006)
21. Wu, Q.X., McGinnity, T.M., Maguire, L.P., Belatreche, A., Glackin, B.: Adaptive Co-Ordinate Transformation Based on Spike Timing-Dependent Plasticity Learning Paradigm. In: Wang, L., Chen, K.S., Ong, Y. (eds.) ICNC 2005. LNCS, vol. 3610, pp. 420–428. Springer, Heidelberg (2005)
22. Jeffress, L.A.: A Place Theory of Sound Localization. *J. Comp. Physiol. Psychol.* 41, 35–39 (1948)
23. Jeffress, L.A.: Binaural Phase Difference and Pitch Variation. *Am. J. Psychol.* 61, 468–486 (1948)
24. Jeffress, L.A.: Interaural Phase Difference and Pitch Variation: Day-to-Day Changes. *Am. J. Psychol.* 62, 1–19 (1949)

On Determination of Early Warning Grade Based on AHP Analysis in Warranty Database

SangHyun Lee, MinTae Lee, Culhyun Kim, ChulSu Park, SeungBeom Park, Yuyang Liu, and ByungKi Kim

Department of Computer Engineering, Chonnam National University, Korea
Leesang64@gmail.com, lightwire@naver.com, mkichhy@nate.com,
av3616@chonnam.ac.kr, puru21@msn.com, Yangmayicomein@126.com,
bgkim@chonnam.ac.kr

Abstract. A warranty claims information system is in general developed as three steps. The first step is a quantitative analysis through the time series detection of warranty claims data. The second step is early warning grade determination, considering both of the quantitative analysis and qualitative factors related to the early warning one. The third step is unit list sampling with pure warranty claims in all these activities. Especially, the considerations in early warning grade determination are the qualitative factors such as change due to complaints of customers, variations between regions, unit types and models, parts significance and so on. AHP analysis is appropriate in connection with these problems. This paper suggests a neural network learning model in determining early warning grade of warranty claims data, which includes AHP analysis and knowledge of quality experts. The early warning grade of warranty claims data using this model can compromise a dispute with rapid quality improvement and cost efficiency. The test result also suggests that the proposed method enhances accuracy of early warning grades in warranty claims database, which is at national famous automobile company.

Keywords: warranty claims; time series detection; AHP analysis; parts significance; neural network; Quality Information Report (QIR).

1 Introduction

The ultimate goal of warranty claims information system lies in the extraction of unit warranty claims list being practical problems. This process is generally composed of three stages: The first stage is the development of modules which distinguishes between the commonality and uniqueness in constituting units. This module uses operation code as key variable, and determines unit warranty claims type. The second stage is the decomposition of the operation code, related to contribution rates of each unit and extraction of core unit number. The analysis of variance is used to the division of commonality and uniqueness, in which the unit number is a key variable.

The third stage is to link together the operations such as the internal relationship between unit types and models based on the unit number system, deciding whether

service agents have trouble or not, warranty claims grade inference considering the unit significance and the warranty claims early-warning system.

According to the study of Pizzi and Pedrycz(2003) on the software quality evaluation, it is desirable that, in the construction of warranty claims information system, the expansion and maintenance of warranty claims early-warning system is needed to be properly applied to this system using fuzzy relation concept as to quartile for the sake of processing various qualitative factors [4]. Grzegorzewski and Hryniewicz attempt to make fuzzy theory-like approach which can model ambiguous data in that, in relation to product reliability, colloquially expressed malfunction information and partially described defects frequently appear in the real world [5].

Ali and Chen(2005) suggests the neural network model which can map process measured value in light of product quality in that, in the problem of product quality, varied different approaches are needed according to its goals.[3] This suggested model emphasizes that those issues occurring in the process of unit properties and quality control can be processed. The problem is that the constraint of neural network structure is strenuous, especially the limitation that input neurons are constricted to 3.

Montis et al. (2000) presents the application standard of NAIDE, MAUT and MOP/GP, and the merits and demerits of their methods as a result of the case study on 4 MCDA(Multi-Criteria Decision Aid) methods.[1]

Zhu et al.(2005) makes a representative study related to quality and AHP(Analytic Hierarchy Process) technique, especially one on various alternatives of structure design about diverse quality properties.[2] This paper is intended to minimize the warning errors resulting from the warranty claims early-warning list extraction by AHP analysis which can naturally quantify the qualitative information in the construction of warranty claims information system. Thus, we propose neural network-based decision-making model which is given knowledges from experts in unit quality. It reduces the warranty claims early-warning error arising from the warranty field. Finally, we apply practically the proposed model to a national car company, thereby reducing time and cost loss by quality control activities.

2 Warranty Claims DB Design for AHP Analysis

2.1 File Construction of Time Series Detection

There are generally two methods with respect to warranty claims time series identification. One is to calculate warranty claims by the standard of production date, and the other is to do by the standard of repair date. The former is favorable to monitoring quality in light of production control because the warranty claims index is calculated by the production date, whereas its difficulty is that it is too complex to judge quality abnormality and the detection time comes relatively late in comparison with the way to be calculated by means of occurrence date. "Time Series Detection" is to complement this which monitors the warranty claims rate by means of repair date. This analysis is the main module in the warranty claims information system.

AHP analysis can take into account the external qualitative things in connection with the abnormality in the warranty claims time series. In particular, there are variation from the seasonality of units, one among operational lines which are due to the

workers' skill and locality errors in the case of bringing about specific region's caution warranty claims. AHP analysis is meaningful in the analysis of variance between unit types and models.

2.2 AHP DB Design

The DB design for AHP analysis is as in Figure 2. It can be designed as the sector starting warranty claims master through customers' information files, warranty claims unit files of service regions to AHP data collective file and that to AHP data file through vehicle master summary. Unit significance can be made up of respective files, including unit division, unit number code, malfunction type, customers' complaint number, warranty-related service information, significance rate etc.

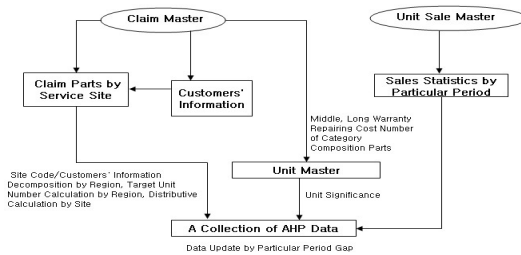


Fig. 1. DB Design for AHP Analysis

Given the set of AHP data, DB design for extracting warranty claims early-warning list is possible. The information file for problem improvement involves presentation contents/time, improvement contents/date, unit model/code, treatment contents, record of line application time. early-warning list is composed of caution time, model code, similar parts group system, parts name code system, service time, unit name, early-warning rate, records of early-warning marks. For your information, warranty claims error rate is calculated as a result of the cause unit number of warranty service (repair) divided by the number of group sales in question. And the part number defined as cause unit can be put together to be part number code system and similar parts group system and the error rate among groups is calculated by means of analysis table.

3 Neural Network Learning Based on AHP Analysis

[Fig. 2] is how to minimize the warning errors resulting from the warranty claims in the early-warning list. It represents the multilayer perceptron structure using 5 input neurons, which are extracted by AHP analysis which can naturally quantify the qualitative information. Sigmoid can be used as a transfer function, and the final grade judgement is adjusted the output neuron value by using the parts significance rate as a weight. Parts significance is the process of quantifying the significance of unit kinds in the level of part code.

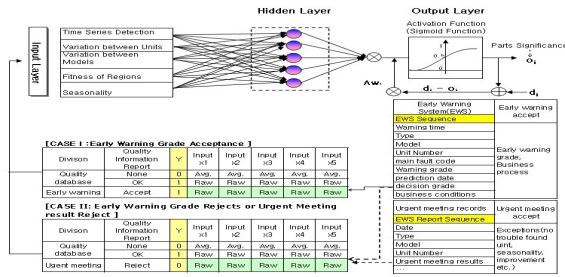


Fig. 2. Neural Network Learning and Weighted Degree Decision

The items according to parts significance can be categorized as three kinds of groups, that is, (1)unit serious degree (nonconforming type, unit type), (2)customer information, (3)warranty repair information. The learning step of neural networks is as follows:

[Step 1] Determine the initial value (learning rate $c=0.05$, gain $\beta=0.0001$, the initial weight multiplies the value of random number generator by 0.5 and the initial threshold also multiplies the random number by 0.5).

[Step 2] Calculate the network output, n (multiply the initial weigh matrix and input vector, add the threshold value).

[Step 3] Calculate the transfer function and its differentiation:

$$f(n) = (1 / (1 + \exp(-n))) / (1 - [f(n)]^2)$$

[Step 4] Calculate the difference between target (d) and neuron output:

$$\delta = (d - f(n)) \times f'(n)$$

[Step 5] Update the weight and threshold by δ :

$$W^{new} = W^{old} + c \times \delta \times X_i, \theta^{new} = \theta^{old} + \beta \times \delta$$

[Step 6] Repeat learning and determine final weight.

4 Case Study

In order to apply AHP analysis related to the construction of warranty claims information system and neural network presented in this study, the warranty claims about A car type of national korea Company is used.

Table 1. AHP Analysis Results of Variation between Unit

Division	Correlation Pattern	Interpretation	Risk	
Common Unit	case 1	High correlation between car types and models	Independently of the properties of car type and model, warranty claims occurs	Most high
	case 2	High in case of car types low among models	In case the properties among car types are great or temporary phenomena	Normal
	case 3	Low in case of car types high among models	In case the properties among models are great or temporary phenomena	Normal
	case 4	Low in the correlativity between car types and models	Temporary phenomena	Low
Unique Unit	case 1	High correlativity between car types and models	Temporary phenomena or data error	Low
	case 2	High in case of car types low among models	Temporary phenomena	Normal
	case 3	Low in case of car types high among models	Temporary phenomena	Normal
	case 4	Low in the correlativity between car types and models	Causes Specific Units (Verification of Data Alignment)	Most high

[Table 1] is the AHP analysis results of variation between units revealed according to the relationship properties between car types and models. AHP analysis of variation between model defines the correlations among models as based on the model index divided into an engine and a body type. The correlation is written out in the same way as the correlation coefficient calculation of car types. But, in the case of car industry, the selection procedure of model combination is needed for correlativity coefficient calculation.

Table 2. Goodness of fit test for repair regions

Region	Warranty claims Number	Estimation Number	Goodness of fit	Region	Warranty claims Number	Estimation Number	Goodness of fit
Seoul	13	15	0.27	Gyeonggi	3	11	5.82
Incheon	4	3	0.33	Gangwon	4	2	2.00
Chungnam	2	2	0.00	Daejeon	1	1	0.00
Chungbuk	3	2	0.50	Busan	4	4	0.00
...				...			

The AHP analysis of regional fitness is reflected in rating by detecting the regional errors (esp. maintenance shops). [Table 2] is the result of goodness of fit test on 16 servicing regions including Seoul

Table 3. A car type Repair-ing Cost Rating

Range of Warranty Service Cost	Rating	Mark
0 ~ 27,138	H	0.7
27,138 ~ 57,672	G	0.8
57,672 ~ 88,207	F	0.9
88,207 ~ 118,742	E	1
118,742 ~ 149,276	D	1.2
149,276 ~ 179,811	C	1.6
179,811 ~ 210,345	B	1.8
210,345 ~ or more	A	2

Table 4. Composite Part Significance

Category	Correlation Coefficient	Relevance	Reference
Engine	0.82	High	[Category High in Relevance]
Seat	0.81	High	-Engine>seat>transmission
Transmission	0.90	High	[Category Low in Relevance]
Thermostat	0.64	Middle	-Thermostat, exterior
Exterior	0.63	Middle	[Category in No Relevance]
Traveling/Steering/Brake System	0.34	Low	-convenience, interior, traveling/ steering /brake system, sound system
Convenience Device	0.28	Low	
Interior	0.29	Low	
Sound System	-	None	

The internal relationship between repairing cost and warranty claims is necessary for the AHP analysis of part significance. The quantification of repairing cost is determined through the opinion of company part expert as in [Table 3], the value which divides warranty claims numbers(C) in case of n=3 months in the short term and n=12 months in the long term by sales number(S) is applied for the quantification of warranty claims ration. [Table 4] is the composite result of part significance. [Table 5] is the result table of the final rating considering the neural network learning using MATLAB version 7.1 and part significance. The final rating result reveals that S, A, and B grades are S grade and the grade under C is A as the best grade respectively. In reality, the grade settlement of early strict parts is substantially followed by frequently unnecessary improvement activities of parts not belonging to the grade in question. But, the application result of the proposed technique brought about very near result similar to quality experts inside the company. Ultimately, the result of remarkably reducing the improvement cost is given birth by considerably reducing

Table 5. Part Significance and Neural Network Learning Rating

Initial Part Significance		Result of AHP-Neural Network	
Rating	Weight	Rating	Weight
S	1.5	S	1.3~
A	1.4		
B	1.3		
C	1.2		
D	1.1	A	1.0~1.2
E	1		
F	0.9		
G	0.8	B	0.8~0.9
H	0.7		
I	0.6	C	0.5~0.7
J	0.5		
		None	0.5 or under

the error warning about parts which don't substantially need improvement in this improvement activity.

5 Conclusions

Now, the difficulty problem in developing warranty claims information is the qualitative parts such as the interrelation among units and models, service (repair) attributes, customer satisfaction and unit significance, etc. In particular, these parts must be taken into consideration in extracting the early-warning list from the warranty claims database. Until now the determination of strict warning grade by the crisp logic has been unacceptable to a quality expert's viewpoint according to the exclusion of the qualitative factors unable to quantify in the unit quality improvement activities. This study presented the preprocess method through AHP analysis able to involve these factors in the warning list extraction and introduced neural network which can be reflected the knowledge of quality experts. Also, as a result of applying the proposed model aiming at the national famous car company, the warranty claims early-warning grade was enormously reduced and showed alignment with the opinion of quality experts in car units in comparison with not applying AHP analysis data, with the good result of minimizing economic loss arising from the warranty service in the company's side.

References

1. Ali, Ö.G., Chen, Y.T.: Design Quality and Robustness with Neural Networks. *Software Quality Journal* 13, 357–375 (2005)
2. Zhu, L., Aurum, A., Gorton, I., Jeffery, R.: Tradeoff and Sensitivity Analysis in Software Architecture Evaluation Using Analytic Hierarchy Process. *Software Quality Journal* 13(4), 357–375 (2005)
3. Montis, A.D., Toro, P.D., Droste-Franke, B., Omann, I., Stagl, S.: Criteria for Quality Assessment of MCDA Methods. In: 3rd Biennial Conference of the European Society for Ecological Economics, Vienna (May 2000)
4. Kunene, K., Petkov, D.: Task Structuring a Brainstorming Group Activity with an AHP-Based Group Support System. In: HICSS: 35th Annual Hawaii International Conference on System Sciences (HICSS 2002), vol. 8, p. 221. IEEE Computer Society, Washington (2002)
5. Douglas, J.M., Butler, J., Mullarkey, P.W.: An Approach to Ranking and Selection for Multiple Performance Measures. In: Winter Simulation Conference, pp. 719–726 (1998)

Optimizing the Performance of Probabilistic Neural Networks Using PSO in the Task of Traffic Sign Recognition

LunBo Li and GuangFu Ma

Department of Control Science and Engineering, Harbin Institute of Technology,
Harbin, Heilongjiang 150001, China
{lunboli, magf}@hit.edu.cn

Abstract. This paper presents a fast version of probabilistic neural network model for the recognition of traffic signs. The model incorporates the J-means algorithm to select the pattern layer centers and Particle Swarm Optimization (PSO) to optimize the spread parameter, enhancing its performance. In order to cope with the degradations, the Combined Blur-Affine Invariants (CBAs) are adopted to extract the features of traffic sign symbols without any restorations which usually need a great amount of computations. The experimental results indicate that the fast version of PNN optimized using PSO is not only parsimonious but also has better generalization performance.

Keywords: probabilistic neural networks; particle swarm optimization; traffic signs recognition; combined blur-affine invariants.

1 Introduction

The idea behind the driving assistance systems (DAS) is to increase security on our roads. Traffic sign recognition is a subtask of DAS which purpose is to hint a driver when specific signs are spotted, for instance to limit speed of his or her car [1]. An automatic traffic sign recognition system identifies traffic signs within live colour images captured by a camera and alerts the driver of the traffic signs. The traffic sign recognition is usually composed by two specific stages: the detection of traffic signs in the image and their classification. In this work we pay special attention to the classification stage. Numerous systems, including rule-based systems [2], matching pursuit method [3], statistical learning systems (naïve Bayes [4], Support Vector Machines [5]) and neural networks [6, 7], have been used to perform classification tasks on this problem.

An effective approach for addressing such tasks is the Probabilistic Neural Networks. PNNs were introduced by Specht in 1990 [8]. The Classical PNN can be viewed as an “intelligent memory” since each training pattern is stored as a neuron of the network [9]. PNNs require small training times and produce outputs with Bayes posterior probabilities. These desirable features come at the expense of larger memory requirements and slower execution speed for the prediction of unknown patterns [8].

The performance of a PNN is largely influenced by the spread parameter. In this paper, we propose a fast version of probabilistic neural network model that incorporate the J-means algorithm to select the patterns layer centers and the Particle Swarm Optimization (PSO) algorithm to optimize the spread parameter. J-means, a local search heuristic, has shown good performance in solving the minimum sum of squares clustering problem [10]. PSO is a swarm intelligence optimization algorithm, motivated by the dynamics of socially organized colonies [11], which has proved to be very efficient on a plethora of applications in science and engineering. The proposed fast version of PNN model is applied to the recognition of degraded traffic signs. In order to cope with the degradations, the Combined Blur-Affine Invariants (CBAs) are adopted to extract the features of traffic sign symbols without any restorations which usually need a great amount of computations [12].

The paper is organized as follows. Section 2 of this paper describes the basic concepts of PNNs and PSO, then the proposed approach is presented. A brief introduction of CBAs and their normalization are articulated in Section 3. Testing procedure and corresponding results of the classification for the traffic signs are shown in Section 4. The paper ends with conclusions.

2 Background Theory and the Proposed Approach

2.1 Probabilistic Neural Network

The probabilistic neural network was introduced by Specht [8]. It is a supervised neural network that is widely used in the area of pattern recognition, nonlinear mapping, and estimation of the probability of class membership and likelihood ratios [13]. The standard training procedure for PNNs requires a single pass over all the patterns of the training set [8]. This characteristic renders PNNs faster to train, compared to feedforward neural networks (FNNs).

The structure of a PNN is similar to that of FNNs, although the architecture of a PNN is always limited to four layers; the *input layer*, *pattern layer*, *summation layer*, and *output layer*, as illustrated in Fig.1. Let $X_{i,k} \in \mathbb{R}^n$ be the i th pattern of the training set that belongs to category k , with $i = 1, 2, \dots, M_k$, $k = 1, 2, \dots, K$, where M_k is the size of class k , and K is the number of categories. For each $X_{i,k}$, a neuron in the pattern layer of the network is created. The center of the Gaussian kernel activation function of this neuron is $X_{i,k}$. The output of the specific pattern neuron is connected to the neuron of the summation layer that corresponds to the class, k , in which the training pattern is classified.

An input vector, $X \in \mathbb{R}^n$, is applied to the input neurons, x_i , $i = 1, 2, \dots, n$, and is passed to the pattern layer. The neurons of the pattern layer are divided into K groups, one for each class. The i th pattern neuron in the k th group computes its output using a Gaussian kernel of the form,

$$f_{i,k}(X) = \frac{1}{(2\pi)^{n/2} \det(\Sigma)^{1/2}} \exp\left(-\frac{1}{2}(X - X_{i,k})^T \Sigma^{-1} (X - X_{i,k})\right), \quad (1)$$

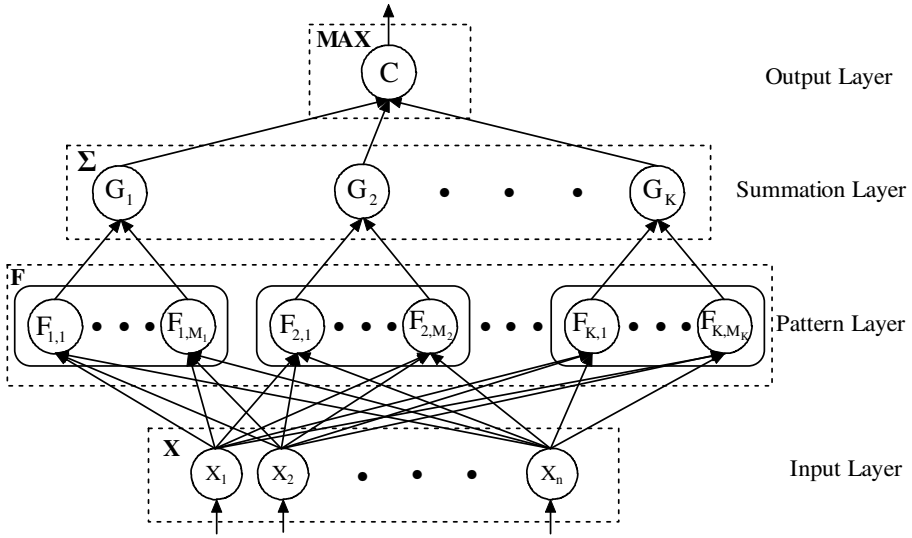


Fig. 1. Structure of the probabilistic neural network

where $X_{i,k} \in \mathbb{R}^n$ is the center of the kernel; Σ also known as the matrix of spread parameters, determines the size and shape of the receptive field of the kernel, and $\det(\Sigma)$ denotes the determinant of the matrix Σ . In his original contribution Specht restricted Σ to one global smoothing parameter, σ^2 . Substituting $\Sigma = \sigma^2 I$, where I is the identity matrix, Eq.(1) becomes,

$$f_{i,k}(X) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{\|X - X_{i,k}\|^2}{2\sigma^2}\right), \tag{2}$$

PNNs that exploit a global smoothing parameter are called homoscedastic. On the other hand, with the term heteroscedastic PNN we refer to networks that use a diagonal matrix Σ instead of a scalar σ^2 [13].

The summation layer of the network computes the approximation of the conditional class probability functions through a combination of the previously computed densities,

$$G_k(X) = \sum_{i=1}^{M_k} w_{ki} F_{k,i}(X), \quad k \in \{1, \dots, K\}, \tag{3}$$

where M_k is the number of pattern neurons of class k , and w_{ki} are positive coefficients satisfying, $\sum_{i=1}^{M_k} w_{ki} = 1$. Pattern vector X is classified to belong to the class that corresponds to the summation unit with the maximum output,

$$C(X) = \arg \max_{1 \leq k \leq K} (G_k). \tag{4}$$

Note that the prior class probabilities, w_k , are problem-dependent, and it may not always be feasible to estimate them from the training data since the training set may contain little meaningful information regarding the prior class probabilities. In our experiments, we assume that all classes are equiprobable, i.e.,

$$w_k = 1/K, \quad 1 \leq k \leq K .$$

2.2 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a stochastic, population-based optimization algorithm. It exploits a population of individuals to synchronously probe promising regions of the search space. In this context, the population is called a swarm and the individuals (i.e. the search points) are called *particles*. Each particle moves with an adaptable velocity within the search space, and retains a memory of the best position it ever encountered. In the *global* variant of PSO, the best position ever attained by all individuals of the swarm is communicated to all the particles at each iteration. In the *local* variant, each particle is assigned to a neighborhood consisting of prespecified particles. In this case, the best position ever attained by the particles that comprise the neighborhood is communicated among them [11].

Assume an d -dimensional search space, $S \subset \mathbb{R}^d$, and a swarm consisting of NP particles. The i -th particle is an $Z_i = (z_{i1}, z_{i2}, \dots, z_{id})^T \in S$. The velocity of this particle is also a d -dimensional vector, $V_i = (v_{i1}, v_{i2}, \dots, v_{id})^T \in S$. The best previous position encountered by the i -th particle in S is denoted by, $BP_i = (bp_{i1}, bp_{i2}, \dots, bp_{id})^T \in S$. Assume g_i to be the index of the particle that attained the best previous position among all the particles in the neighborhood of the i -th particle, and t to be the iteration counter. Then the swarm is manipulated by the equations,

$$V_i(t+1) = wV_i(t) + c_1r_1(BP_i(t) - Z_i(t)) + c_2r_2(BP_{g_i}(t) - Z_i(t)) , \quad (5)$$

$$Z_i(t+1) = Z_i(t) + V_i(t+1) , \quad (6)$$

where $i = 1, \dots, NP$, w is a parameter called *inertia weight*; c_1 and c_2 are two positive constants called *cognitive* and *social* parameter, respectively; and r_1 , r_2 , are random variables uniformly distributed within $[0, 1]$. Alternatively, the velocity update can be performed through the following equation [14]

$$V_i(t+1) = \chi[V_i(t) + c_1r_1(BP_i(t) - Z_i(t)) + c_2r_2(BP_{g_i}(t) - Z_i(t))] , \quad (7)$$

where χ is a parameter called *constriction factor*, giving rise to different version of PSO. The best positions are then updated according to the equation

$$BP_i(t+1) = \begin{cases} Z_i(t+1), & \text{if } f(Z_i(t+1)) > f(BP_i(t)) \\ BP_i(t), & \text{otherwise} \end{cases} , \quad (8)$$

and the indices $g_i, i = 1, 2, \dots, NP$, are update, where f is the objective function to be maximized by PSO. In our case f declares the sum of conditional probability of each class membership of the training set. The particles are always bounded in the search space S .

2.3 The Proposed Approach

A fast version of the PNN can be obtained by using only a part of the training data set T instead of the whole training data set. Such a training set L can be obtained by finding some “representatives” of the training data through a clustering technique. In our approach, we identified an adequate number of informative representatives (mean centers) from each class by using the J-means clustering algorithm on the training data of each class [10]. The classification accuracy of a PNN is influenced by the spread parameters of its kernels. In our study, we allow the spread parameters, Σ , to be a diagonal matrix whose entries can differ. Obvious, in the one-dimensional case, Σ becomes σ^2 . We refer to the former case as heteroscedastic PNN, while the latter case is referred to as homoscedastic. For the homoscedastic case the optimization task is in one dimension, whereas for the heteroscedastic case the dimension of the optimization problem equals that of the pattern vectors. The pseudocode for the fast version of PNN is shown in Table 1.

Table 1. Pseudocode for the fast version of PNN

PNN input: Training set T.	
PSO input: Swarm size NP, χ, c_1, c_2, bounding box $S = [0.01, 0.00001]^d$	
Step 1	Apply the J-Means algorithm on each class of the training set T to obtain the clustered training set L .
Step 2	Perform a pass over the set L to construct the PNN and optimize the spread parameters using PSO. Set $t=0$. Initialize a PSO swarm, $Z_i(t) = \text{diag}(\Sigma_i(t)) \in B, V_i(t) \in B, i = 1, 2, \dots, NP$. Initialize best positions, $BP_i(t), i = 1, 2, \dots, NP$, and the indices g_i . Do Update velocities, $V_i(t+1), i = 1, 2, \dots, NP$, using Eq. (7). Update particles, $Z_i(t+1) = Z_i(t) + V_i(t+1), i = 1, 2, \dots, NP$. Constrain each particle $Z_i(t+1)$ in S . Compute the fitness function: $f(Z_i(t+1))$. Update best positions, $BP_i(t+1), i = 1, 2, \dots, NP$, and the indices g_i . Update iteration counter, $t = t + 1$. While (the maximum number of iteration is not reached) Write the optimal spread parameter.
Step 3	Compute final classification from the proposed approach.

3 Feature Collection — Combined Blur and Affine Invariants

The determination of invariant characteristics is an important problem in traffic sign recognition. In many situations, traffic signs symbols to be processed are usually subjected to geometric distortion and blur degradation. Thus, having combined affine-blur invariants is in great demand. The explicit forms of combined invariants were derived in the paper [12].

As the magnitudes of the different orders of CBAs differ greatly, the CBAs are not proper to be used directly in pattern recognition. It is necessary to normalize the magnitudes of CBAs for the traffic recognition. It can be found that all the moment invariant polynomials are quantic, and they can be defined in a uniform form as follows:

$$I = \left(\sum_{t=1}^n k_t \prod_{j=1}^m \mu_{p_j q_j} \right) / \mu_{00}^l, \quad (9)$$

where m, l , are invariable for each order moment invariants. Once the grey-level of image increase λ times, μ_{pq} increase λ times and the moment invariants I is changed to I / λ^{l-m} . Therefore, a new magnitude normalization method for the moment invariants is proposed as follows:

$$\bar{I} = \text{sgn}(I)^{(l-m)} \sqrt[l-m]{|I|}, \quad (10)$$

where

$$\text{sgn}(x) = \begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases}. \quad (11)$$

The magnitude of each order of normalized moment invariants is proportional to λ^{-1} , and the normalized moment invariants are favorable for the pattern recognition.

4 Experimental Results

In this study, we only take the red and round prohibition signs into account and leave the traffic sign of no-bicycle climbing lane out of account, then get 38 prohibition signs as illustrated in Fig.2.

The prohibition signs in Fig.2 are transformed with 6 different scale transformations and 6 rotation transformations, and we get 1368 samples. Then, we select 18 samples from each class traffic sign, and get 684 training samples. The test sets are designed as follows: Firstly, the 1368 samples are transformed by 7 blur degradation with averaging mask of size $3 \times 3, 5 \times 5, 7 \times 7, 9 \times 9, 11 \times 11, 13 \times 13, 15 \times 15$ pixels, respectively. Then, we get 8 test sets with the original samples. Secondly, the 8 test sets are degraded with additive white noise with STD of 0, 0.01 ... 0.07, respectively, and 64 test sets are generated.



Fig. 2. Standard circular traffic sign symbols

The pattern layer centers of PNN are determined using J-means algorithm and the global smoothing parameter σ is optimized with jackknife method [15]. In this way, we get a fast version of homoscedastic PNN, the hidden centers of which are chosen as about 5.6% (38 hidden centers) the original training samples. The test sets classification accuracy percentage of the fast version of homoscedastic PNN is presented in Table 3. Table 3 shows that the classification accuracy of the classifier is not as good as that of standard homoscedastic PNN in Table 2. In order to improve the classification accuracy and the generalization performance of the classifier, we use the spread parameters, Σ , whose entries differ from each other instead of the global smoothing parameter σ , then optimize the spread parameters using Particle Swarm Optimization. The corresponding test sets classification accuracy percentage is shown in Table 4.

It can be seen from the Table 3 and Table 4 that, the fast version of heteroscedastic PNN has better classification accuracy and generation performance than that of the fast version of homoscedastic PNN. With a deep look between the classification accuracy in Table 2 and that in Table 4, it can be seen that the fast version of heteroscedastic PNN has comparable classification accuracy to that of the homoscedastic PNN and better generation performance.

It needs to be pointed out that, when the averaging mask size is comparable to that of the traffic signs, “boundary effect” prevents from using the blur invariants in such cases. This is the main reason that the classification accuracy declines as the averaging mask increases.

Table 2. Classification accuracy percentage with the standard version of homoscedastic PNN

Mask \ STD	0x0	3x3	5x5	7x7	9x9	11x11	13x13	15x15
0	99.83	99.57	95.12	87.96	80.47	72.89	63.47	52.44
0.01	99.24	97.05	91.75	85.18	80.22	70.62	62.37	51.01
0.02	89.81	87.96	87.71	78.87	74.58	70.20	58.84	48.40
0.03	72.39	78.62	86.03	72.39	68.68	61.61	54.96	43.26
0.04	80.05	70.28	70.20	75.00	59.51	56.73	46.96	39.73
0.05	63.55	61.03	68.35	55.64	51.17	49.57	43.18	35.85
0.06	59.59	53.03	66.41	44.19	46.29	34.93	35.35	35.26
0.07	57.83	46.55	35.43	47.73	37.71	29.12	37.96	30.30

Table 3. Classification accuracy percentage with the fast version of homoscedastic PNN

Mask \ STD	Mask							
	0×0	3×3	5×5	7×7	9×9	11×11	13×13	15×15
0	98.65	98.48	93.18	86.27	78.28	71.12	60.94	50.16
0.01	98.56	95.79	88.88	82.57	76.68	68.68	59.68	48.90
0.02	85.94	87.54	86.27	75.84	72.13	66.49	56.39	46.04
0.03	66.58	77.18	84.09	70.45	65.57	57.99	52.60	42.34
0.04	74.24	71.21	68.85	71.88	55.38	52.44	45.03	38.88
0.05	60.60	56.06	63.38	53.70	48.82	47.30	41.41	34.51
0.06	56.14	47.97	62.28	43.51	43.26	32.74	34.51	34.51
0.07	55.80	46.04	31.98	45.11	37.96	29.04	36.61	29.12

Table 4. Classification accuracy percentage with the fast version of heteroscedastic PNN

Mask \ STD	Mask							
	0×0	3×3	5×5	7×7	9×9	11×11	13×13	15×15
0	98.99	98.82	97.39	92.68	86.11	77.36	68.43	56.73
0.01	98.65	96.46	92.68	91.67	85.10	74.83	66.58	55.56
0.02	90.24	86.36	93.77	83.08	75.51	70.87	62.79	51.52
0.03	80.39	82.32	87.88	80.56	75.51	66.33	59.60	46.38
0.04	86.62	72.81	73.82	80.64	63.89	60.27	51.60	43.60
0.05	68.43	65.91	67.59	56.23	58.92	55.64	43.94	37.54
0.06	64.31	56.82	60.35	45.37	57.91	42.42	38.05	35.35
0.07	54.21	47.05	47.90	52.27	39.98	35.52	37.12	32.58

5 Conclusions

Traffic sign recognition is an important subtask of DAS. During the acquisition of traffic sign symbols, the blur and affine degradations are introduced inevitably which make it difficult for invariant features extraction on traffic signs undergoing degradations. In this paper, the combined blur-affine invariants are adopted to extract the invariant features, and a novel normalization method is proposed to cope with the great difference of magnitude of CBAs.

A fast version of heteroscedastic PNN is proposed which incorporate J-means clustering method to determine the pattern layer centers and PSO to optimize the spread parameter, enhancing its performance. The proposed approach is applied to the recognition of degraded traffic signs, the simulation results indicate that the designed model is not only parsimonious but also has better generalization performance.

References

1. Fletcher, L., Apostoloff, N., Petersson, L., Zelinsky, A.: Vision in and out of Vehicles. *IEEE Intell. Syst.* 18, 12–17 (2003)
2. Paclik, P., Novovicova, J.: Road Sign Classification without Color Information. In: Proc. of the 6th Annual Conference of the Advanced School of Imaging and Computing. Springer, Belgium (2000)

3. Farag, A.A., Abdel-Hakim, A.E.: Detection, Categorization and Recognition of Road Signs for Autonomous Navigation. In: Proceedings of Advanced Concepts in Intelligent Vision Systems, pp. 125–130. Springer, Belgium (2004)
4. Hsu, S.H., Huang, C.L.: Road Sign Detection and Recognition Using Matching Pursuit Method. *Image and Vision Computing* 19, 119–129 (2001)
5. Maldonado-Bascón, S., Lafuente-Arroyo, S., Gil-Jiménez, P., Gómez-Moreno, H., López-Ferreras, F.: Road-Sign Detection and Recognition Based on Support Vector Machines. *IEEE Transactions on Intelligent Transportation Systems* 8, 264–278 (2007)
6. De la Escalera, A., Armingol, J.M., Mata, M.: Traffic Sign Recognition and Analysis for Intelligent Vehicles. *Image and Vision Computing* 21, 247–258 (2003)
7. Nguwi, Y.Y., Kouzani, A.Z.: Detection and Classification of Road Signs in Natural Environments. *Neural Computing & Applications* (2007) DOI: 10.1007/s00521-007-0120-z
8. Specht, D.F.: Probabilistic Neural Networks. *Neural Networks* 1, 109–118 (1990)
9. Berthold, M.R., Diamond, J.: Constructive Training of Probabilistic Neural Networks. *Neurocomputing* 19, 167–183 (1998)
10. Hansen, P., Mladenović, N.: J-Means: A New Local Search Heuristic for Minimum Sum of Squares Clustering. *Pattern Recognition* 34, 405–413 (2001)
11. Kennedy, J., Eberhart, R.C.: *Swarm Intelligence*. Morgan Kaufmann, San Francisco (2001)
12. Suk, T., Flusser, J.: Combined Blur and Affine Moment Invariants and Their Use in Pattern Recognition. *Pattern Recognition* 36, 2895–2907 (2003)
13. Specht, D.F., Romsdahl, H.: Experience with Adaptive Probabilistic Neural Networks and Adaptive General Regression Neural Networks. In: *IEEE International Conference on Neural Networks*, Orlando, FL, pp. 1203–1208 (1994)
14. Clerc, M., Kennedy, J.: The Particle Swarm-Explosion, Stability, and Convergence in a Multidimensional Complex Space. *IEEE Transactions on Evolutionary Computation* 6, 58–73 (2002)
15. Masters, T.: *Practical Neural Network Recipes in C++*, pp. 201–222. Academic Press, San Diego (1993)

Product Schemes Evaluation Method Based on Improved BP Neural Network

Weiwei Chen¹, Xiaopeng Wei^{1,2,*}, and Tingting Zhao¹

¹ Liaoning Key Lab of Intelligent information Processing, Dalian University,
116622 Dalian, China

² Institute of Mechanical Engineering, Dalian University of Technology, 116024 Dalian, China
*xpwei@dlu.edu.cn

Abstract. Improved back propagation (BP) neural network evaluation method for product schemes took the main index data as input vector, took the sample comprehensive scores as output by using the analytic hierarchy process (AHP). The network was separately trained by momentum factorial algorithm, Gauss–Newton algorithm and Levenberg-Marquardt algorithm. With the application and verification in Haier refrigerator schemes, the comparison of speed and mean absolute error show that the BP neural network trained by Levenberg-Marquardt algorithm is reliable.

Keywords: BP; AHP; momentum factorial algorithm; Levenberg-Marquardt algorithm; Gauss–Newton algorithm.

1 Introduction

With the globalization of economic development, competition among enterprises is increasingly fierce. The key point of business survival depends on whether the product can satisfy customer requirements. Generally speaking, the conceptual design phase will usually have numbers of schemes. In order to improve design efficiency, enhance success rate, reduce cost and shorten design cycles, the product schemes should be evaluated and the best one should be put into produce.

Product schemes evaluation is a multi-factor, complex and nonlinear system. In order to show its result actually, the dynamic and nonlinear problem must be settled in the process of evaluation. At present, there are many evaluation methods [1], such as Analytic Hierarchy Process, Fuzzy Comprehensive Evaluation method and Grey Correlation Analysis, these methods possess rationality and feasibility, but they can not settle the above problem, and their evaluation efficiency is low.

The BP neural network [2] provides a new effective way of evaluation for product schemes with the characteristic of non-linear approach, the stronger ability of adaptation, studying, proceeding together and reasoning logically. For this, the evaluation method is put forward based on the self-study function of the BP neural network. It weakens the random and appraiser's subjectivity when fixing the index weight in evaluation.

2 BP Neural Network and Improved Algorithms

The back propagation (BP) algorithm, the most typical multi-layer feed forward neural network, is one of the most widely used algorithms. BP neural network is basically a gradient descent algorithm to minimize the error function in the weights space.

But the gradient descent method has three main inherent problems: being prone to getting stuck in a local minimum, having the slow convergence speed and lacking stable generalization ability in the applications. Many researchers have made deep improvements on the basis of basic BP algorithm, such as adding momentum to gradient descent algorithm[3], Gauss–Newton (G-N) algorithm[4] and Levenberg-Marquardt (L-M) [4]algorithm.

Let $V(\vec{w})$ denote the manifestation function of the network, where \vec{w} is the weight on the connection, we have to adjust \vec{w} to minimize the value of $V(\vec{w})$. Using the G-N method, the correction of weight $\Delta\vec{w}$ can be expressed in the form:

$$\Delta\vec{w} = -[\nabla V^2(\vec{w})]^{-1} \nabla V(\vec{w}). \quad (1)$$

Where $\nabla V^2(\vec{w})$ and $\nabla V(\vec{w})$ represent the Hessian matrix and the gradient. Assume that the manifestation function $V(\vec{w})$ is calculated by summing the error squares of neurons in the output layer. It can be defined as:

$$V(\vec{w}) = \sum_{i=1}^N e_i^2(\vec{w}). \quad (2)$$

Where $e_i^2(\vec{w})$ denotes the error signal of the j th output neuron. Furthermore, we can find the formulas as follows:

$$\nabla V(\vec{w}) = J^T(\vec{w})\vec{e}(\vec{w}), \nabla^2(\vec{w}) = J^T(\vec{w})J(\vec{w}) + S(\vec{w}). \quad (3)$$

Where $J(\vec{w})$ represents the Jacobian matrix and

$$S(\vec{w}) = \sum_{i=1}^N e_i(\vec{w})\nabla^2 e_i(\vec{w}). \quad (4)$$

For the G-N method, it is normally assumed that: $S(\vec{w}) \approx 0$. The G-N method is modified by L-M method with:

$$\Delta\vec{w} = [J^T(\vec{w})J(\vec{w}) + \mu I]^{-1} J^T(\vec{w})\vec{e}(\vec{w}). \quad (5)$$

Where μ is a scalar quantity. By adjusting the parameter μ , the L-M algorithm can maneuver between its two extremes: the gradient descent ($\mu \rightarrow \infty$) and the G-N algorithm ($\mu \rightarrow 0$). According to the result of train iterations, the L-M algorithm

adjusts the parameter μ dynamically to change the direction of convergence. Calculating the Jacobian matrix plays a key role in the algorithm. When the number of weights in the network is small, the L-M algorithm can improve the learning speed effectively.

3 Improved BP Neural Network on Evaluation

3.1 Structure of the Neural Network

In principle, it has been proved that every neural network model with only one hidden layer can uniformly approximate any continuous function [5]. So a three-layer BP network is employed in our study. The BP neural network m neurons in the input layer according to the number of evaluation indices, one neuron in the output layer according to sample comprehensive scores and n neurons in the hidden layer which are determined by empirical formula [6]:

$$n = \sqrt{n_1 \times n_2} + a. \quad (6)$$

Where n_1 and n_2 are the numbers of input and output knot, and a is in range of 1~10. We choose $a = 4$.

The transfer function between input layer and hidden layer is Sigmoid function:

$$f(x) = \frac{1}{1 + e^{-x}}. \quad (7)$$

and the transfer function between hidden layer and output layer is purelin function.

3.2 The Input and Output of the Neural Network

3.2.1 The Input

The entire input variable must be normalized before training BP neural network: on one hand, the dimensions of different indices are usually different, we can't compare the difference between evaluation object's attribute with different dimensions directly; on the other hand, transfer function of the hidden layer nodes is hyperbolic tangent, large data absolute value will make great influence on the transmission effect of the function [7], thus affecting the training results of neural networks.

First, we must scale the attributes that denote by nature language, we may take 0.8 for "good", 0.6 for "normal", and 0.4 for "bad". And then, for all the data, we scale them into [0, 1]. For the benefit indices (the bigger the better):

$$Y_j = \frac{X_{ij} - \min_j}{\max_j - \min_j}. \quad (8)$$

and the cost indices (the smaller the better):

$$Y_j = \frac{\max_j - X_{ij}}{\max_j - \min_j}. \quad (9)$$

3.2.2 The Output

Sample comprehensive scores reflect the quality of schemes, the calculate formula is as follow:

$$SCS = \sum_{i=1}^N W_i Y_i. \quad (10)$$

Where W_i represents index weight and Y_i is non dimensional processed data. We get W_i by using Analytic Hierarchy Process (AHP), detail in [8].

3.3 Training the Neural Network

For the non-linear system, choosing initial weights is very important. It will directly influence the convergence of learning and the time of the training. If the initial weight is too big, the weighted output will fall in the saturation of Sigmoid function and the adjustment of the weights will be near zero which will result in the halt of learning process. We hope that the primly weighted output of each neuron will be near zero so that each neuron weight could make adjustment in the widest rang of Sigmoid activation function. Normally, the initial weights will be random in (-1, 1) [9].

Set“net.trainParam.show=5,net.trainParam.epochs=2000,net.trainParam.goal=1e-5”.Training the BP neural network respectively with adding momentum to gradient descent algorithm, G-N algorithm and L-M algorithm. The training samples are inputted to the system and Matlab neural network tool box with the *trainidx*, *trainbfg* and *trainlm* study methods were used to train the samples.

3.4 Test and Preserve the Neural Network

Select several sample put into the net, compare between training output and expectation output, if the relative error can be accepted preserve the weights and thresholds between layers.

4 Example

Choose Haier refrigerator schemes evaluation for example.

4.1 Refrigerator Evaluation Indices

We extract ten main indices of customer requirement through market research as evaluation indices, shown in Table 1. The cost indices are X2, X5 and X6, the rest are benefit indices. We get index weight $W_i = (0.0754 \ 0.1384 \ 0.1384 \ 0.2302 \ 0.0384 \ 0.0754 \ 0.0709 \ 0.0454 \ 0.1469 \ 0.0406)$.And the original data is shown in Table 2.

Table 1. Refrigerator evaluation indices system

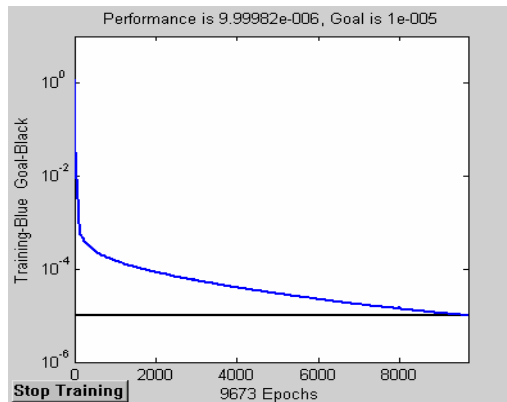
Target layer	Indices layer	Index unit
Refrigerator evaluation indices system	Effective Volume (X1)	L
	Power Consumption (X2)	KWh/24h
	Input Power (X3)	W
	Fresh-keeping Ability (X4)	Kg/24h
	Weight (X5)	Kg
	Noise (X6)	Db
	Environmental Protection (X7)	0-1
	Modeling effect (X8)	0-1
	Life (X9)	Year
	Measure (X10)	0-1

Table 2. The original data of schemes

Scheme	1	2	3	4	...	35	36
X1	70	72	70	72	...	117	117
X2	0.5	0.5	0.52	0.52	...	0.71	0.7
X3	80	85	90	95	...	130	130
X4	0.8	0.83	0.90	0.93	...	0.9	1.1
X5	25	25	27	27	...	37	37
X6	32	31	32	34	...	36	37
X7	0.62	0.89	0.64	0.78	...	0.87	0.83
X8	bad	normal	normal	good	...	bad	good
X9	8	8	10	11	...	15	16
X10	bad	normal	bad	bad	...	bad	good

4.2 Training and Comparison

The training set is the first 28 schemes in Table 2, data of another 8 are selected as the test sample. We trained the BP neural network respectively with adding momentum to

**Fig. 1.** Performance of momentum factorial algorithm

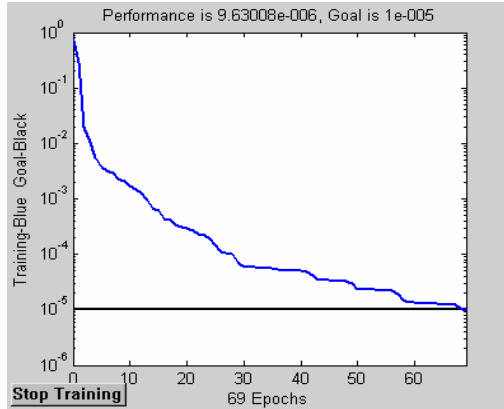


Fig. 2. Performance of G-N algorithm

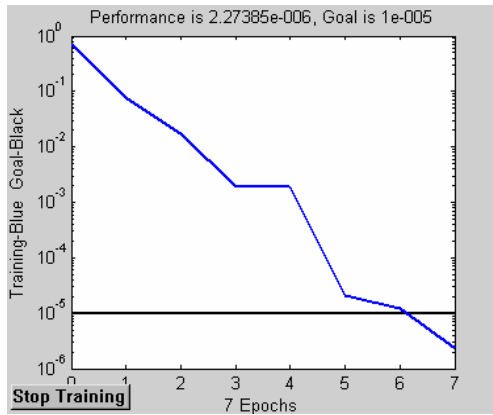


Fig. 3. Performance of L-M algorithm

Table 3. The epoch and mean relative error of the three algorithms

Algorithm	Epoch	Mean relative error(%)
The momentum factorial	9673	2.27
G-N	69	3.93
L-M	7	0.28

gradient descent algorithm, G-N algorithm and L-M algorithm and the performances of three algorithms are indicated in Fig. 1, Fig. 2 and Fig. 3.

Table 3 shows the epoch and mean relative error of the three algorithms, it is proved that the BP neural network trained by L-M algorithm is the best in speed and training effect. And the test results have shown in Table 4.

Table 4. Test Results

Item No.	29	30	31	32	33	34	35	36
Training output	0.5220	0.4180	0.5047	0.5167	0.5848	0.3717	0.4951	0.5998
Expectation output	0.5216	0.4176	0.5038	0.5177	0.5853	0.3747	0.4987	0.5993
The absolute value of relative error(%)	0.0325	0.0366	0.0873	0.0998	0.0477	0.3011	0.3586	0.0496

Preserve the weights and the thresholds of the net which are trained by L-M algorithm, shown in Table 5 and Table 6. It can be used to evaluate refrigerator schemes by inputting non dimensional processed data of evaluation indices.

Table 5. The weights and thresholds between input layer and hidden layer after training

	1	2	3	4	5	6	7
1	0.2539	0.1053	0.3729	-0.3628	-0.2471	0.1155	0.9619
2	0.5071	0.7008	-0.4023	0.0747	0.6802	0.3354	-0.5965
3	0.1552	-0.2467	0.1870	-0.2766	0.0095	0.1416	0.3576
4	-0.0916	-0.1428	-1.1234	-0.1672	0.6861	0.3051	-0.0743
5	-0.9141	-0.4108	-0.0712	-0.4280	-0.1847	-0.0238	0.9183
6	-0.0980	-0.4315	-0.6953	0.1136	-0.7854	0.0235	-0.0272
7	-0.3138	-0.3146	-0.5534	0.7623	-0.8265	0.0647	0.8808
8	0.3546	0.0147	0.4106	-0.7091	-0.2352	0.0396	0.6346
9	-0.2449	-0.0072	-0.0004	-0.3096	-0.2839	0.2100	0.1561
10	0.0192	0.0564	1.0277	-0.3349	0.4323	0.0768	-0.0236
Threshold	-0.7402	0.1620	0.2339	0.7101	0.3443	-0.6210	0.0539

Table 6. The weights and thresholds between hidden layer and output layer after training

1	2	3	4	5	6	7	Threshold
0.0287	-0.1320	-0.0130	-0.0321	-0.0075	0.6648	0.0022	0.4772

All of the analysis above are based on $a = 4$ in formula 6, that means we have 7 neurons in hidden layer. And when we change the parameter a , the L-M algorithm is always the best performance. Owing to space constraints, not list in detail.

5 Conclusions

In this paper, the BP neural network trained by Levenberg-Marquardt optimization algorithm has better generalization performance, less training steps and more reliable results. After training, it can be used to simulate the process of comprehensive evaluation by experts. Combined with knowledge acquisition and expert system and it can be form a knowledge base for scheme evaluation.

It extricates us from fussy programming by using Matlab neural network tool box and enables us to concentrate on schemes evaluation. Therefore, it is of great significance by using Matlab neural network.

Acknowledgments. Thanks for the support of National Natural Science Foundation, China (No.50575026) , Excellent Youth Talents Foundation of Liaoning Province, China (No.3040014), Education Office of Liaoning Province, China (No.20060039 and No.2005063).

References

1. Liu, Z.W., Deng, S.E., Teng, H.F.: Survey on the Evaluation Methods of Design Schemes in a Complicated Engineering System. *J. Systems Engineering and Electronics* 25(12), 1488–1491 (2003)
2. Ju, C.H., Liang, Y., Liu, D.S.: A kind of E-government Website Evaluation Method Based on Neural Network. In: *The 3th IEEE International Conference on e-Business Engineering (ICEBE 2006)*, pp. 407–414. IEEE Press, New York (2006)
3. Schraudolph, N.N.: Gradient-based Manipulation of Nonparametric Entropy Estimates. *J. IEEE Transactions on Neural Networks* 15(4), 828–837 (2004)
4. Ye, F., Zhou, G.G., Lu, J.Q.: The Risk-Evaluation Model in Customs Based on BP Neural Networks. In: *IEEE International Conference on Natural Computation (ICNC 2007)*, pp. 181–184. IEEE Press, New York (2007)
5. Zheng, B.X., Chen, G.M.: A Comprehensive Evaluation Method of Safety at Oil Depot Based on Artificial Neural Network. *J. Industrial Engineering and Management* 10(2), 13–19 (2004)
6. Xiao, D.Y., Wang, Z.J., Chen, R.D.: Performance Evaluation of Strategy Analysis and Bargaining Mechanism Based on Neural Network Training. *J. Industrial Engineering and Management* 2, 70–75 (2005)
7. Kong, Y., Liu, L.: The Method Research about the Supplier Appraises Based on BP Neural Network. *J. Value Engineering* 26(6), 89–92 (2007)
8. Jiang, W.D.: Evaluation of R&D Personnel's Competence based on AHP and BP Neural Network in the Enterprise. *J. Systems Engineering-Theory&Practice* 25(12), 1488–1496 (2003)
9. Xia, D.Y.: Fire Risk Evaluation Model of High-rise Buildings Based on Multilevel BP Neural Network. In: *4th IEEE International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)*, pp. 436–441. IEEE Press, New York (2007)

Study on Tool Wear Monitoring Based on Multi-source Information Fusion

Lanshen Guo^{*}, Haiwei Zhang^{*}, Yanxia Qi^{*}, and Zhi Wei^{*}

School of Mechanical Engineering, Hebei University of Technology,
Tianjin, 300130, China
guolanshen@163.com

Abstract. The paper developed a system of tool wear monitoring in advanced manufacture systems. In conventional wear-monitoring method, it cannot exhibit unique behavior found in regular modern machining systems. Because a single monitoring signal, such as the signal of force, temperature, ultrasound or AE, cannot exactly describe the state of tool work for monitoring in advanced manufacture. This paper, therefore, mainly researched on real-time cutter state monitoring using neural network, neural network integration and multi-sensor information integrating technology. Picture pattern-recognition and feature extracting were adopted and combined with other information of the cutter dynamically. The characteristic information was gathered using an appropriate model of cutter wear or damage. Neural network were used to imitate the complicated nonlinear mapping relationship and to fuse multi-kind sensors that collect wearing and damage information and make decision and judgment rapidly.

Keywords: On-line; Tool wears monitoring; Multi-source Information fusion; Neural network integration; Decision-making.

1 Introduction

On-line tool condition monitoring was essential for modern machining systems, especially in the case of precession and unmanned machining [1]. Among the various applications for machine intelligence, monitoring the machining process was characterized as the most imperative. Original observation method--through eyes should be improved through an automatic monitoring system. Performances of the monitoring system have direct influence on products quality and production rate in modern machining systems. It was known that the tool flank wear in cutting process also had a major influence on the subsurface quality of the work-piece, but crater wear governs the reliability of the operation because crater wear would eventually result in the fracture of the tool. Diffusion and chemical reactions were responsible for the crater wear in cutting process. Collision and tool breakage were sudden and mostly unexpected

^{*} The authors would like to thank **Science Technology R&D Guidance Program in Hebei Province** (Grant No: 05212173), and supported by the funds for PhD teachers, Hebei University of Technology.

events that require reactions in real time, but the development of tool wear was gradual. The importance of tool wear monitoring was also implicit in the possible economic advantages. If worn tools could be changed rapidly, it would be possible to avoid the waste of production. Furthermore, tool costs could be reduced noticeably by an optimized exploitation of the tool life. An accurate estimation of tool wear would even make it possible to adjust the tool position in order to meet geometric specifications and to guarantee a certain surface quality of the work-piece [1],[2].

Tool state monitoring method was composed of direct and indirect monitoring. The direct monitoring was directly observing the tool state to judge whether the tool breakage was happened, and the most typical method was Industrial Television (ITV). The indirect monitoring was observing physics quantity or physics phenomena to judge indirectly whether the tool breakage was happened or would be happened. The methods included measuring force method, measuring temperature method, measuring vibration method, measuring main motor method, and measuring acoustic emission method. It was difficult to dispose the single sensor measured signal caused by certain fault, because the signal always hid in other background. Literature [3] presented multi-sensor fusion method so as to make up lower reliability of single signal.

Multi-sensor fusion technology aimed to achieve the overall measuring information at the same measuring targets, and by different treatment methods. Therefore, it was necessary to improve measure precision and dependability. In multi-sensor systems, information appeared in variety, complexity and large capacity. Information process differed from single sensing measure and the corresponding technology, and multi-sensor merge technology had become an important research field. [3],[4]

In conventional wear-monitoring method the single monitoring signal cannot accurately describe the state of tool work processes which was used for monitoring tools, such as force signal, temperature signal ,ultrasound waves and AE(Acoustic Emission)[5],[6],[7]. The main task of this paper focused on real-time cutter state monitoring, which applied neural network, neural network integration, multi-sensor information fusion technology, and adopted picture pattern-recognition, feature extracting with dynamical cutter in machining process. In order to obtain dynamical and comprehensive monitoring information of cutting tool, and improve anti-jamming ability, the sub-monitoring network needed to be set up separately, which gathered cutting tool condition information from the different aspect and adopted the precise decision-making to the fusion nerve network. Through the decision and judgment ability of neural network that could imitate the complicated non-linear mapping relationship, the wearing and damaged information from multi-source of sensors were rapidly fused. In order to design an accurate tool wear monitoring system, various aspects associated with monitoring tool system were investigated in this study.

2 Application of Neural Network Integration

2.1 Establishment of Single BP Fuzzy Neural Network

Of multi-sensor systems, system working information offered by each sensor had uncertainty in certain degree. The integration course to this uncertain information

was, in fact, an uncertain reasoning process. Because of the separation for original characteristic between fuzzy logic and neural network technologies, fuzzy technology and neural network can be combined to make fuzzy neural network control system to extract fuzzy rule and to create fuzzy membership function automatically, and online regulation. In literature [8], flank wears was predicted by using back propagation neural network. In literature [9], BP fuzzy neural network construction method was adopted to merge information of each individual BP neural network, which did not depend on the accurate mathematic model of the system and was suitable for complicated system and course.

The constructed individual BP neural network structure was illustrated in Fig.1. When the fuzzy study method on the study mechanism of the individual network was adopted, the characteristic of this kind of learning method was that the learning process was measured with the fuzzy quantity (degree of membership), i.e., the input volume was the fuzzy quantity after being fuzzy. Suppose a fuzzy neural network had one input and two outputs with training set (X_1, T_1) , where $X_1 = (X_{11}, X_{12})$ and T_1 was the expected output. If each fuzzy quantity was adopted to be triangle membership function (the same for other membership function) $X_{1,j} \in [1,0]$ and right value $T_i \in [-1,1]$, the error signal was:

$$E = \frac{1}{2} \sum_{i=1}^L (T_i - Y_i)^2 \quad (1)$$

The purpose of learning was to minimize E, but $Y_i = T_i$, E will not be zero, because of the particularity of the fuzzy system. Therefore, according to different demands, different stopping iteration rules were needed. The fuzzy rule of multi-input multi-output system can be divided into some fuzzy rule of multi-input single output system.

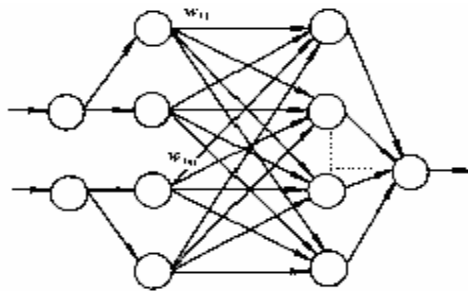


Fig. 1. A back propagation fuzzy neural network

2.2 Integration of Two Multi-layer BP Disposal Networks

Fig. 2 shows that the signal disposal neural network dealt with the signal of sensor measuring, and then extracted useful information, which will be an input to fusion neural network. In order to get more comprehensive and accurate information, fusion

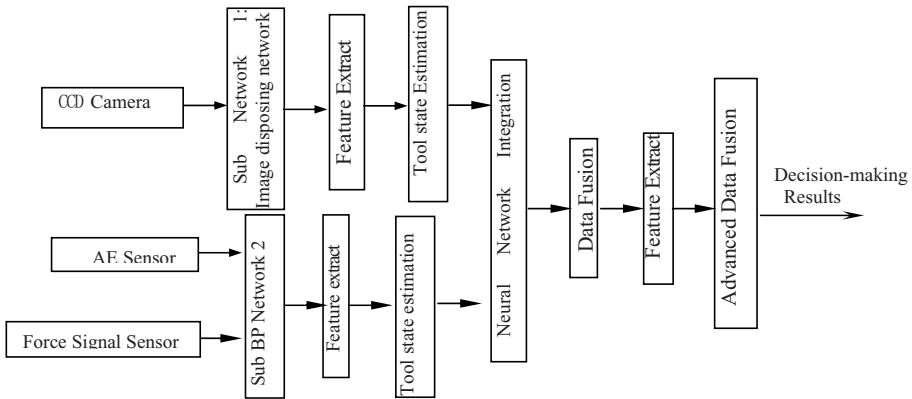


Fig. 2. The flowchart of Multi-source sensor fusion

neural network can take order to inputs on the definite phase. It can put up two sub-networks of disposing different sensors separately (shown in Fig. 1), then combined each individual network by way of parallel integration, thus, got a high-powered recognition system. Each BP network unit can be accomplished finely to image disposing, dimension cutting down and feature extracting. Every part of sensor’s target vector X was corresponding to every input of BP neural network. The individual network learned the date of training set. From test set A and B to test the test result was regarded as an input of the integrated network. The integrated network individual was produced by Boosting methods.

3 Multi-sensor Fusion System Using Neural Network Integration

This multi-sensor fusion system is shown in Figure 3. It was composed of multi-source sensor group such as the CCD camera, AE, and force signal sensor; image acquisition card; image processing software; input and output devices; etc. In the system, Sub-network 1 and Sub network 2 gathered signal of wearing tool at the same time. This system used the outputs of each sensor as the inputs of Sub-network and used the output of Sub-network as the input of Integration Neural Network. In Sub-network 1 the CCD camera worked in a given period interval to obtain the image of cutter, and then the system went on analyzing and processing the image, dimensionality reduction and feature extracting. V_B was the wear extent of flank, and regarded it as feature vector. CCD camera can obtain a two-dimensional gray level array (namely image) in tool cutting. In this case, the dimension to measure the space at this moment was too high to be suitable for classifying machine with design. The measuring space with huge dimensions must be transferred into the characteristic space with dimensions greatly cut down, in which the studied image can be expressed by characteristic vector. This was the course of characteristic extraction.

In Sub-network 2, AE sensor and force signal sensor gather signal of wearing tool to eliminate or reduce various kinds of noise caused by insufficient light, vibration,

white noise, etc. The main advantage of the AE was its independence from the cutting direction. The feed and the thrust direction forces should be separated for most of the monitoring applications by using the cutting forces. AE sensor was sensitive to microscopic characteristic of cutting process (for example plastic deformation and friction in the cutting area), and cutting force sensor was sensitive to cutting tool and work piece vibration that caused by cutting tool attrition [7]. Therefore, the method of using AE and cutting force sensor to monitor cutting tool attrition can obtain microscopic information (stress wave) and macroscopic information (vibration). Thus some effective characteristic signals can be gather to realize monitoring cutting tool wears.

The system can acquire some states of tool signal, which cannot be measured by the traditional sensors. In Integration Neural Network, Sub-network 1 and Sub-network 2 integrate in the form of parallel integration. Compared with the standard characteristic information stored on the computer in advance, it would fuse advanced data in advanced layer. If the result was true, the controlling system would signal an alarm message to change this cutter. Compared with standard value, system can judge whether the cutter is qualified. The schematic diagram of multi-sensor fusion system was shown in Fig. 3

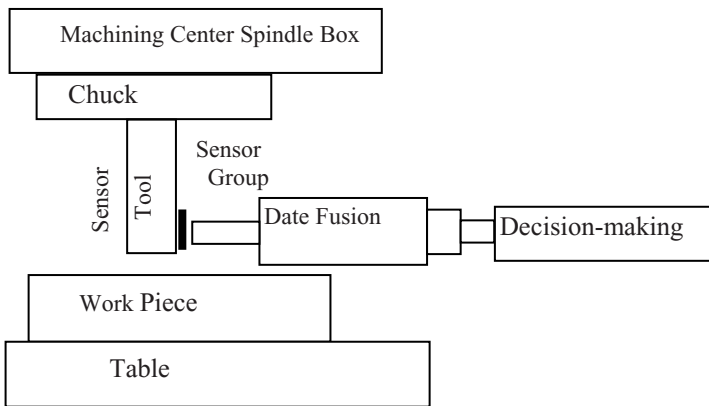


Fig. 3. The system schematic diagram

4 Experiments

In this section, we applied the multi-sensor fusion system for a vertical machining center numerical to estimate tool state. The cutting processed cutting parameter were: $a_p = (0.1 \sim 0.3 \text{ mm})$, $f = (8 \sim 35) \text{ mm/min}$, $n = (500 \sim 800) \text{ r/min}$, work piece material was 45# steel. The cutting tool was the HSS end mill with vertical handle and three edges. The sampling frequency of AE was 5MHz, Sampling length 2,048. The sampling frequency of cutting force signal sensor was 1 kHz, Sampling length 1,024. The experiment carried through 6 kind of cutting tools conditions, respectively were normal cutting ($< 0.1 \text{ mm}$), mild attrition ($0.1 \sim 0.3 \text{ mm}$), moderate attrition ($0.3 \sim 0.6 \text{ mm}$) and serious attrition ($0.6 \sim 0.8 \text{ mm}$), blade breakage ($1.0 \sim 1.2 \text{ mm}$), blade

Table 1. Correspondence between cutting tool condition and goal output of neural network

Tool state	Mild attrition	Moderate attrition	Serious attrition	Blade breakage	Blade falling
Normal cutting	010000	001000	000100	000010	000001
Target output					
100000					

falling (over 1.4 mm). Each kind of condition gathered 20 groups of data, each group of data length was 2,048, randomly chose 10 groups for training sample, in the remainder randomly chose 5 groups for testing sample. Table 1 showed cutting tool condition and nerve network goal output correspondence relations.

After selecting the suitable starter value and the deviation, and then the network were trained through 227 times of circulation trainings, error sum of squares of network fell on in 0.05, and saved successful weight and the deviation into the memory. NN tested using 20 testing samples that had trained, judged the cutting tool state, result like Table 2 shows.

Table 2. Network output and target output of testing sample

Tool states	Actual outputs of neural network integration						Target output
Normal cutting	0.8651	0.0051	0.0021	0.0013	0.0009	0.0001	100000
Mild attrition	0.0050	0.7814	0.0067	0.0035	0.0010	0.0000	010000
Moderate attrition	0.0023	0.0046	0.6765	0.0038	0.0007	0.0002	001000
Serious attrition	0.0019	0.0039	0.0051	0.8245	0.0024	0.0003	000100
Blade breakage	0.0046	0.0052	0.0048	0.0032	0.7459	0.0001	000010
Blade falling	0.0034	0.0047	0.0058	0.0029	0.0038	0.6989	000001

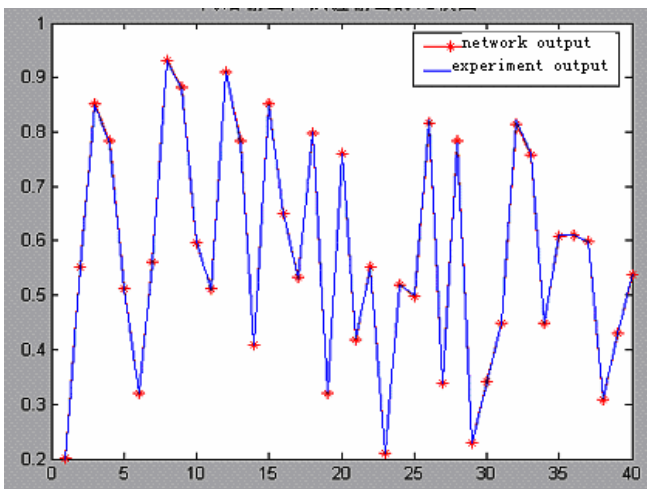


Fig. 4. The comparison of network output and experiment output

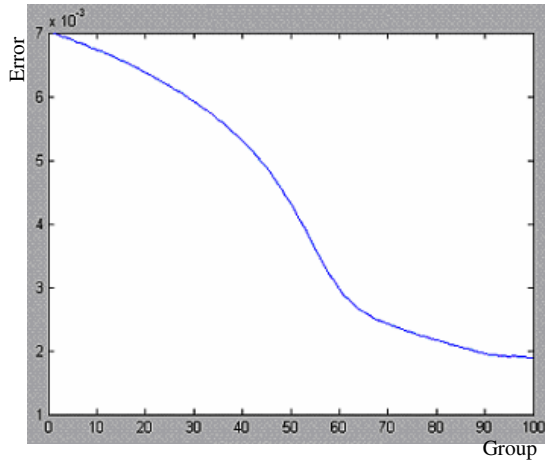


Fig. 5. The error in training process

Draw a conclusion from table 2: the actual situation was consistent with tool states estimated by actual outputs of neural network. These results indicated the cutting tool monitor system using neural network integration was feasible. Simulate the work process of system using Matlab, obtained the curves of the practical output and the experiment output, validated the validity of using neural network to fuse Sub BP network 1 and 2, using neural network integration.

Fig.4 showed the comparison of network output and experiment output, the fusion result of two sub networks was correct, and Fig.4 also showed simulation for the Multi-source Information Fusion system was presented and discussed in this paper. The result that the integrated network exports indicates a good identification.

Show as the Fig.5, because it was all joined some noises in the neural network study and the training process, it existed some slight errors in the network output and in the actual output comparison. The network output and the experiment output basically tallied, the error controlled in expectation error in network training process.

5 Conclusion

In this paper a new system is proposed to detect tool breakage and to estimate tool wear states. The proposed system uses CCD image, AE signals and force signal after they were processed to inspect cutting tool states. Two kinds of subnets are put up, trained and tested. The results indicate that the system can predict the wear values accurately, with the decision-making error of the predictions at 5% despite the noisy nature of the training and testing data. The approximate shape of the flank and crater wear can be consequently determined in the system which requires low cost monitoring system. Tool breakages in all the tested cases are detected by the system. The system is proved reliable and response fast compared with the simple computation, which does not take processor time.

References

1. Nidal, H.A.Z., Yu, G.: Analytical Model for Tool Wear Monitoring in Turning Operations Using Ultrasound Waves. *International Journal of Machine Tools & Manufacture* 40, 1619–1635 (2000)
2. Scheffer, C., Kratz, H., Heyns, P.S., Klocke, F.: Development of A Tool Wear-Monitoring System for Hard Turning. *International Journal of Machine Tools & Manufacture* 43, 973–985 (2003)
3. Di, Y., Dany, E.W.T.I.: A Multi-sensor Strategy for Tool Failure Detection in Milling. *Int. J. Mach. Tools Manufact.* 35(3), 383–389 (1995)
4. Sick, B.: Online and Indirect Tool Wear Monitoring in Turning with Artificial Neural Networks: A Review of More Than a Decade of Research. *Mechanical Systems and Signal Processing* 16, 487–546 (2002)
5. Scheffer, C., Heyns, P.S.: Wear Monitoring in Turning Operations Using Vibration and Strain Measurements. *Mechanical Systems and Signal Processing* 15, 185–202 (2002)
6. Lee, B.Y., Liu, H.S., et al.: Monitoring of Tool Fracture in End Milling Using Induction Motor Current. *Journal of Materials Processing Technology* 70, 279–284 (1997)
7. Tansel, M.T., Nedbouyan, A.: Micro-end-milling—III. Wear Estimation and Tool Breakage Detection Using Acoustic Emission Signals. *International Journal of Machine Tools & Manufacture* 38, 1449–1466 (1998)
8. Ozel, T., Nadgir, A.: Prediction of Flank Wear by Using Back Propagation Neural Network Modeling When Cutting Hardened H-13 Steel with Chamfered and Honed CBN Tools. *International Journal of Machine Tools & Manufacture* 42, 287–297 (2002)
9. Kothamasu, R., Huang, S.H.: Intelligent Tool Wear Estimation for Hard Turning: Neural-Fuzzy Modeling and Model Evaluation. In: *Proceedings of the Third International Conference on Intelligent Computation in Manufacturing Engineering*, Ischia, Italy, pp. 343–346 (2002)

Synchronization and Parameter Identification for a Class of Chaotic Neural Networks with Time-Varying Delays Via Adaptive Control

Zhongsheng Wang¹, Yanjun Liang², and Nin Yan¹

¹ College of Automation, Guangdong Polytechnic Normal University, Guangzhou, P.R. China, 510635

² College of Information Science and Engineering, Ocean University of China, Qingdao, China 266071

Abstract. The paper aims to present a new synchronization and parameter identification scheme for a class of time-varying neural networks. By combining the adaptive control method and the Razumikhin-type Theorem, a novel delay-independent and decentralized linear-feedback control with appropriate updated law is designed to achieve the synchronization and parameter identification. The updating law of parameters can be directly constructed. Hopfield neural networks with time-varying delays are given to show the effectiveness of the presented synchronization scheme.

1 Introduction

In recent years, due to neural networks' promising potential for the tasks of classification, associate memory and parallel computation, communication such as secure communication through the chaotic system, etc., it has attracted the attention of the scientists. Those neural networks have been applied to describe complex nonlinear dynamical systems, and have become a field of active research over the past two decades [1-15]. It is known that the finite speed of amplifiers and the communication time of neurons may induce time delays in the interaction between the neurons when the neural networks were implemented by very large-scale integrated (VLSI) electronic circuits. Many researchers have devoted to the stability analysis of this kind of neural networks with time-delays. The chaotic phenomena in Hopfield neural networks and cellular neural networks with two or more neurons and differential delays have also been found and investigated [11-15]. Neural networks are nonlinear and high-dimensional systems consisting many neurons. In this paper, the decentralized control method is discussed for the synchronization problem of a class of chaotic systems. By combining the adaptive control method and the Razumikhin-type Theorem, a delay-independent and decentralized linear-feedback control with appropriate updated law is designed to achieve the Globally exponential synchronization. The controllers and the regulating law of parameters can be directly constructed. Hopfield neural networks with time-varying delays are given to show the effectiveness of the presented synchronization scheme.

2 Synchronization Problem Formulations

In this paper, we consider the chaotic neural networks with time-varying delay described by the differential delayed equation of the form

$$\dot{x}_i(t) = -c_i x_i(t) + \sum_{j=1}^n a_{ij} f_j(x_j(t)) + \sum_{j=1}^n b_{ij} g_j(x_j(t - \tau_j(t))) + J_i, i=1, 2, \dots, n, \quad (1)$$

Or in a compact form

$$\dot{x}(t) = -Cx(t) + Af(x(t)) + Bg(x(t - \tau(t))) + J \quad (2)$$

Where $n \geq 2$ denotes the number of neurons in the networks, x_i is the state variable associated with the i neurons, $c_i x_i(t)$ is an appropriately behaved function remaining the solution of chaotic neural networks (1) bounded. The feedback matrix $A = (a_{ij})_{n \times n}$ and the delayed feedback matrix $B = (b_{ij})_{n \times n}$ indicate the interconnection strength among neurons without and with time-varying delays $\tau_j(t) \geq 0$ respectively. The activation function f_i, g_i describe the manner in which the neurons respond to each other, J_i is an external constant input, it is assumed that $0 \leq \tau_j^* = \max(\tau_j(t))$ for $i = 1, 2, \dots, n$ and $t \geq 0$. The initial conditions of system (1) are given by $x_i(t) = \varphi_i(t) \in C([- \tau_j^*, 0], R)$, $C([- \tau_j^*, 0], R)$ denotes the set of all continuous functions from $[- \tau_j^*, 0]$ to R , $C = \text{diag}(c_1, c_2, \dots, c_n) > 0$;

$$f(x(t)) = (f_1(x_1(t)), f_2(x_2(t)), \dots, f_n(x_n(t)))^T;$$

$$g(x(t - \tau(t))) = (g_1(x_1(t - \tau_1(t))), g_2(x_2(t - \tau_2(t))), \dots, g_n(x_n(t - \tau_n(t))))^T$$

$$J = (J_1, J_2, \dots, J_n)^T;$$

We refer to model (1) as the drive chaotic neural networks, and the response neural networks is given by the following equation

$$\dot{z}_i(t) = -\hat{c}_i z_i(t) + \sum_{j=1}^n a_{ij} f_j(z_j(t)) + \sum_{j=1}^n b_{ij} g_j(z_j(t - \tau_j(t))) + J_i + \varepsilon_i e_i(t), i=1, 2, \dots, n, \quad (3)$$

Or in a compact form

$$\dot{z}(t) = -\hat{C}z(t) + Af(z(t)) + Bg(z(t - \tau(t))) + J + \varepsilon \otimes e(t) \quad (4)$$

Where $e(t) = (e_1(t), e_2(t), \dots, e_n(t))^T = (z_1(t) - x_1(t), z_2(t) - x_2(t), \dots, z_n(t) - x_n(t)) \in R^n$ ($i=1, 2, \dots, n$) denotes the synchronization errors, $\hat{C} = \text{diag}(\hat{c}_1, \hat{c}_2, \dots, \hat{c}_n) > 0$ is the estimated

values of the parameters C , $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n) \in R^n$, ($i = 1, 2, \dots, n$) is the updated feedback gain, and the mark \otimes is defined as $\varepsilon \otimes e(t) = (\varepsilon_1 e_1(t), \varepsilon_2 e_2(t), \dots, \varepsilon_n e_n(t))^T$. The initial conditions of system (3) are given by $z_i(t) = \psi_i(t) \in C([- \tau_j^*, 0], R)$. Furthermore, we define estimation error of the parameter C as $\tilde{c}_i = c_i - \hat{c}_i, i = 1, 2, \dots, n$.

The error dynamical between system (2) and (4) can be expressed by the following equation

$$\dot{e}(t) = \tilde{C}z(t) - Ce(t) + A\tilde{f}(e(t)) + B\tilde{g}(e(t - \tau(t))) + \varepsilon \otimes e(t) \tag{5}$$

Where

$$\begin{aligned} \tilde{f}(e(t)) &= f(e(t) + x(t)) - f(x(t)), \\ \tilde{g}(e(t - \tau(t))) &= g(e(t - \tau(t)) + x(t)) - g(x(t)) \end{aligned}$$

Before proceeding, an assumption regarding f_i, g_i , is given below.

Assumption 1. Each neuron activation function in (1) $f_i, g_i, i = 1, 2, \dots, n$ is bounded and satisfies the Lipschitz condition with the Lipschitz constants k_i, h_i , that is

$$|f_i(u) - f_i(v)| \leq k_i |u - v|, |g_i(u) - g_i(v)| \leq h_i |u - v| \tag{6}$$

for all $u, v \in R$.

The aim of this paper is to design the updated law of parameters \hat{C} to achieve the synchronization between system (1) and (3) and the parameter identification, that is:

$$\lim_{t \rightarrow \infty} e_i(t) = \lim_{t \rightarrow \infty} (c_i - \hat{c}_i) = 0 \text{ for all } i, j = 1, 2, \dots, n$$

3 Main Results

Main Theorem. For system (1) and (3) which satisfy **Assumption 1**, if the updated feedback gain $\varepsilon_i \in R$ ($i = 1, 2, \dots, n$), and the estimated parameters $\hat{C}, \hat{A}, \hat{B}$ are updated according to the following updated law respectively

$$\dot{e}_i = -\eta_i e_i^2(t), i = 1, 2, \dots, n \tag{7}$$

$$\dot{\hat{c}}_i = \sigma_i e_i(t) z_i(t) \tag{8}$$

where $\eta_i > 0, \sigma_i > 0 (i = 1, 2, \dots, n)$ are arbitrary constants, respectively, then the synchronization and the parameter identification of system (1) and (3) can be carried out.

Proof. In order to confirm the origin of (5) with updating laws (7) and (8) is globally exponentially stable, we construct the Lyapunov function V as

$$V = \frac{1}{2} e^T(t) e(t) + \frac{1}{2} \sum_{i=1}^n \frac{1}{\sigma_i} \tilde{c}_i^2 + \frac{1}{2} \sum_{i=1}^n \frac{1}{\eta_i} (\varepsilon_i + l)^2 \tag{9}$$

Where $l > 0$ is a constant to be determined.

Calculating the derivative of (9) along the trajectories of (5), we obtain

$$\dot{V} = e^T(t) \dot{e}(t) - \sum_{i=1}^n \tilde{c}_i e_i(t) z_i(t) - \sum_{i=1}^n (\varepsilon_i + l) e_i^2(t)$$

Since

$$\begin{cases} e^T(t) \tilde{C} z(t) = \sum_{i=1}^n \tilde{c}_i e_i(t) z_i(t), \\ e^T(t) (\varepsilon \otimes e(t)) = \sum_{i=1}^n \varepsilon_i e_i^2(t) \end{cases}$$

We have

$$\begin{aligned} \dot{V} &= -e^T(t) C e(t) + e^T(t) A \tilde{f}(e(t)) + e^T(t) B \tilde{g}(e(t - \tau(t))) - \sum_{i=1}^n l e_i^2(t) \\ &\leq -e^T(t) C e(t) + \|e^T(t) A\| \|\tilde{f}(e(t))\| + \|e^T(t) B\| \|\tilde{g}(e(t - \tau(t)))\| - \sum_{i=1}^n l e_i^2(t) \\ &\leq -e^T(t) C e(t) + \frac{1}{2} e^T(t) A A^T e(t) + \frac{1}{2} \tilde{f}^T(e(t)) \tilde{f}(e(t)) + \frac{1}{2} e^T(t) B B^T e(t) \\ &\quad + \frac{1}{2} \tilde{g}^T(e(t - \tau(t))) \tilde{g}(e(t - \tau(t))) - \sum_{i=1}^n l e_i^2(t) \end{aligned} \tag{10}$$

From (6) yields

$$\tilde{f}^T(e(t)) \tilde{f}(e(t)) = \sum_{i=1}^n \tilde{f}_i^2(e_i(t)) \leq \sum_{i=1}^n k_i^2 e_i^2(t) \leq k e^T(t) e(t) \tag{11}$$

$$\tilde{g}^T(e(t)) \tilde{g}(e(t)) = \sum_{i=1}^n \tilde{g}_i^2(e_i(t)) \leq \sum_{i=1}^n h_i^2 e_i^2(t) \leq h e^T(t) e(t) \tag{12}$$

where $k = \max\{k_i^2 \mid i = 1, 2, \dots, n\}$, $h = \max\{h_i^2 \mid i = 1, 2, \dots, n\}$

Substituting inequality (11) and (12) in to (10) yields

$$\dot{V} \leq e^T(t) [-C + \frac{1}{2} A A^T + \frac{1}{2} k I + \frac{1}{2} B B^T - l I] e(t) + \frac{1}{2} h e^T(t - \tau(t)) e(t - \tau(t))$$

According to Razumikhin-type Theorem 4.1 in [16], if $\|e(t - \tau(t))\| \leq \alpha \|e(t)\|$ ($\alpha > 1$ is a constant) then

$$\begin{aligned} \dot{V} &\leq e^T(t) \left[-C + \frac{1}{2}AA^T + \frac{1}{2}kI + \frac{1}{2}h\alpha^2 I + \frac{1}{2}BB^T - lI \right] e(t) \\ &\leq [\lambda_{\max}(-C + \frac{1}{2}AA^T + \frac{1}{2}kI + \frac{1}{2}h\alpha^2 I + \frac{1}{2}BB^T) - l] e^T(t)e(t) \end{aligned}$$

Where I is an $n \times n$ identity matrix and $\lambda_{\max}(M)$ denotes the maximal eigenvalues for symmetric matrix M .

Taking $l = \lambda_{\max}(-C + \frac{1}{2}AA^T + \frac{1}{2}kI + \frac{1}{2}h\alpha^2 I + \frac{1}{2}BB^T) + 1$,

We can obtain

$$\dot{V} \leq -e^T(t)e(t) \tag{13}$$

Therefore, the solution of (5) with (7) and (8) in about equilibrium point $e(t) = 0$ and $c_i = \hat{c}_i, i = 1, 2, \dots, n$ are globally uniformly stable. Then $e(t)$ and $\hat{c}_i, i = 1, 2, \dots, n$ are globally bounded for $t \geq 0$, vector $e(t)$ is square-integrable by inequality (13), from Eq.(6) $\dot{e}(t)$ is bounded. By Barblat's lemma [17], we conclude that $e(t) \rightarrow 0$, as $t \rightarrow +\infty$, that is, $\lim_{t \rightarrow +\infty} e(t) = 0$.

Moreover, from Eq.(5), we can easily know that $\ddot{e}(t)$ is bounded, that is $\|\ddot{e}(t)\| \leq K$ ($K > 0$ is constant). Suppose $\overline{\lim}_{t \rightarrow \infty} \dot{e}(t) \neq 0$, then there exists an infinite unbounded sequence $\{t_n\}$ and $\varepsilon > 0$ such that $\|\dot{e}(t_n)\| \geq \varepsilon$. For any $t > t_n$, we have

$$e(t) = e(t_n) + \dot{e}(t_n)(t - t_n) + \frac{1}{2} \ddot{e}(t')(t - t_n)^2$$

where $t' \in (t_n, t)$, then

$$\begin{aligned} \|e(t)\| &= \left\| e(t_n) + \dot{e}(t_n)(t - t_n) + \frac{1}{2} \ddot{e}(t')(t - t_n)^2 \right\| \\ &\geq \left\| \dot{e}(t_n)(t - t_n) + \frac{1}{2} \ddot{e}(t')(t - t_n)^2 \right\| - \|e(t_n)\| \end{aligned}$$

Therefore

$$\begin{aligned} \|e(t)\| + \|e(t_n)\| &\geq \left\| \dot{e}(t_n)(t - t_n) + \frac{1}{2} \ddot{e}(t')(t - t_n)^2 \right\| \geq \|\dot{e}(t_n)\|(t - t_n) - \left\| \frac{1}{2} \ddot{e}(t')(t - t_n)^2 \right\| \\ &\geq \varepsilon(t - t_n) - \frac{K}{2}(t - t_n)^2 = (t - t_n) \left[\varepsilon - \frac{K}{2}(t - t_n) \right] \end{aligned}$$

Taking $t = t_n + \delta$, then

$$\|e(t)\| + \|e(t_n)\| \geq \delta \left(\varepsilon - \frac{K}{2} \delta \right)$$

Choosing $\delta = \frac{\varepsilon}{K}$, we have

$$\|e(t)\| + \|e(t_n)\| \geq \frac{\varepsilon^2}{2K},$$

Hence, we get a contradiction with $e(t) \rightarrow 0$ as $t \rightarrow +\infty$, this implies $\overline{\lim}_{t \rightarrow \infty} \dot{e}(t) = 0$. In the same way we have $\underline{\lim}_{t \rightarrow \infty} \dot{e}(t) = 0$. That is $\lim_{t \rightarrow \infty} \dot{e}(t) = 0$. From Eq.(5), we conclude that $\lim_{t \rightarrow \infty} (c_i - \hat{c}_i) = 0$. This completes the proof.

4 Illustrative Example

Example. Consider the following Hopfield neural networks as in [14]

$$\dot{x}_i(t) = -c_i x_i(t) + \sum_{j=1}^2 a_{ij} f_j(x_j(t)) + \sum_{j=1}^2 b_{ij} f_j(x_j(t - \tau_j(t))), \quad i = 1, 2 \quad (14)$$

Where

$$c_i = 1, A = (a_{ij})_{2 \times 2} = \begin{bmatrix} 2 & -0.1 \\ -5 & 2 \end{bmatrix}, B = (b_{ij})_{2 \times 2} = \begin{bmatrix} -1.5 & -0.1 \\ -0.2 & -1.5 \end{bmatrix}, \quad \text{and}$$

$f_i(x_i(t)) = \tanh(x_i)$, respectively. The delays $\tau_1(t) = \tau_2(t) = (1 - \cos t)$.

Remark 1. In our example, the time-varying delays $\tau_1(t), \tau_2(t)$ must not be satisfied $\dot{\tau}_i(t) < 1, (i = 1, 2)$

To achieve synchronization, the response system is designed as

$$\dot{z}_i(t) = -\hat{c}_i z_i(t) + \sum_{j=1}^2 a_{ij} f_j(z_j(t)) + \sum_{j=1}^2 b_{ij} f_j(z_j(t - \tau_j(t))) + \varepsilon_i (z_i(t) - x_i(t)), \quad i = 1, 2 \quad (15)$$

and the updating law of parameters are

$$\dot{\varepsilon}_i = -\eta_i \varepsilon_i^2(t), \quad i = 1, 2 \quad (16)$$

$$\dot{\hat{c}}_i = \sigma_i \varepsilon_i(t) z_i(t), \quad i = 1, 2 \quad (17)$$

then the system (14) and (15) can be globally exponential synchronization. Fig.1-Fig 2 depict the synchronization error of the state variables and the estimation errors

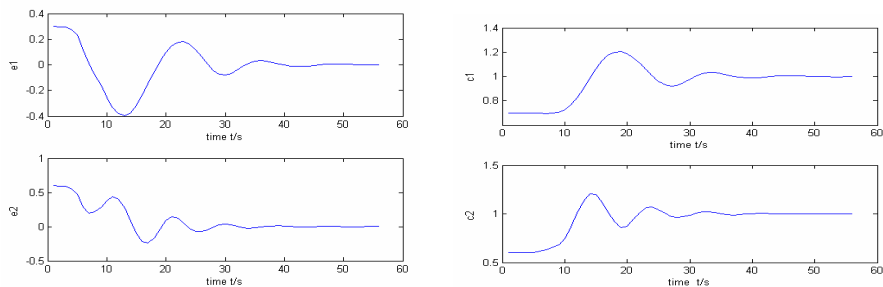


Fig. 1. Synchronization errors for $\tau_1(t)=\tau_2(t)=(1-\cos t)$ **Fig. 2.** The estimations of parameters C_i

of parameter between the drive system(14) and the response system (15) with $\eta_i = 1, \sigma_i = 0.5(i, j = 1, 2)$, and the initial condition $[x_1(s), x_2(s)]^T = [0.4, 0.3], [z_1(s), z_2(s)]^T = [0.1, -0.3]$, respectively.

5 Conclusion

The synchronization and the parameter identification problem for a class of neural networks with time-varying delays has been discussed in the paper. A delay-independent and decentralized linear-feedback control with appropriate updated law is designed to achieve the globally exponential synchronization. The updating law of parameters can be directly constructed. Hopfield neural networks with time-varying delays are given to show the effectiveness of the presented synchronization scheme.

Acknowledgement

This work is supported by National Natural Science Foundation of China (No.60774051), and Foundation of science research of Guangdong polytechnic Normal University.

References

1. Liang, X.B., Wu, L.D.: Globally Exponential Stability of Hopfield Neural Networks and Its Applications. *Sci. China (series A)* 25, 523–532 (1995)
2. Forti, M., Tesi, A.: New Conditions for Global Stability of Neural Networks with Application to Linear and Quadratic Programming Problems. *IEEE Trans. on Circ. and Sys. I: Fundamental Theory and Application* 42, 354–366 (1995)
3. Liao, X.X., Xiao, D.M.: Globally Exponential Stability of Hopfield Neural Networks with Time-Varying Delays. *ACTA Electronica Sinica* 28, 1–4 (2000)
4. Marco, M.D., Forti, M., Tesi, A.: Existence and Characterization of Limit Cycles in Nearly Symmetric Neural Networks. *IEEE Trans. on Circ. and Sys. I: Fundamental Theory and Application* 49, 979–992 (2002)

5. Forti, M.: Some Extensions of a New Method to Analyze Complete Stability of Neural Networks. *IEEE Trans. on Neural Networks* 13, 1230–1238 (2002)
6. Zeng, Z.G., Wang, J., Liao, X.X.: Global Exponential Stability of a General Class of Recurrent Neural Networks with Time-Varying Delays. *IEEE Trans. on Circ. and Sys. I: Fundamental Theory and Application* 50, 1353–1358 (2003)
7. Zeng, Z.G., Wang, J., Liao, X.X.: Global Asymptotic Stability and Global Exponential Stability of Networks with Unbounded Time-Varying Delays. *IEEE Trans. Circuits Syst. II: express briefs* 52, 168–173 (2005)
8. Zeng, Z.G., Wang, J.: Improved Conditions for Global Exponential Stability of Recurrent Neural Networks with Time-Varying Delays. *IEEE Trans. on Neural Networks* 17, 623–635 (2006)
9. Cao, J.D., Huang, D.S., Qu, Y.Z.: Global Robust Stability of Delayed Recurrent Neural Networks. *Chaos, Solitons and Fractals* 23, 221–229 (2005)
10. Fantacci, R., Forti, M., Marini, M., Tarchi, D., Vannuccini, G.: A Neural Network for Constrained Optimization with Application to CDMA Communication Systems. *IEEE Trans. on Circ. and Sys. II: Analog and digital signal processing* 50, 484–487 (2003)
11. Zhou, S.B., Liao, X.F., Yu, J.B.: Chaos and Its Synchronization in Two-Neuron Systems with Discrete Delays. *Chaos, Solitons and Fractals* 21, 133–142 (2004)
12. Cao, J.D.: Global Stability Conditions for Delayed CNNs. *Hopfield Neural Networks and Its Applications. IEEE Trans. On Circuits Syst. I* 48, 1330–1333 (2001)
13. Cheng, C.J., Liao, T.L., Yan, J.J., Hwang, C.C.: Synchronization of Neural Networks by Decentralized Feedback Control. *Physics Letters A* 338, 28–35 (2005)
14. Cheng, C.J., Liao, T.L.: Exponential Synchronization of a Class of Neural Networks with Time-Varying Delays. *IEEE Tran. On systems, man, and cybernetics-part B: Cybernetics* 36, 209–215 (2006)
15. Lu, J.Q., Cao, J.D.: Synchronization-based Approach for Parameters Identification in Delayed Chaotic Neural Networks. *Physica A* 382, 672–682 (2007)
16. Hale, J.: *Theory of Functional Differential Equations*, New York, Heidelberg, Berlin (1977)
17. Krstic, M., Kanellakopoulos, I., Kokotovic, P.: *Nonlinear and Adaptive Control Design*. Wiley, New york (1995)

A Discrete Binary Version of the Electromagnetism-Like Heuristic for Solving Traveling Salesman Problem

Nikbakhsh Javadian¹, Mohsen Gol Alikhani¹, and Reza Tavakkoli-Moghaddam²

¹ Department of Industrial Engineering, Mazandaran University of Science and Technology, P.O. Box: 734, Babol, Iran
{nijavadian, golalikhani}@ustmb.ac.ir

² Department of Industrial Engineering, Faculty of Engineering, University of Tehran, P.O. Box: 11365/4563, Tehran, Iran
tavakoli@ut.ac.ir

Abstract. The electromagnetism-like method (EM) is a meta-heuristic algorithm utilizing an attraction-repulsion mechanism to move sample points (i.e., our solutions) towards the optimality. In general, the EM method has been initially used for solving continuous optimization problems and could not be applied on combinatorial optimization ones. This paper proposes a discrete binary version of the electromagnetism-like method (EM) for solving the combinatorial optimization problems. To show the efficiency of our proposed EM, we use it for solving the traveling salesman problem and compare our computational results with those reported in the literature. Finally we conclude that our method is capable of solving such well-known problems more efficiently than the previous works.

Keywords: Electromagnetism-like method; Traveling salesman problems; Evolutionary computing.

1 Introduction

The electromagnetism-like mechanism (EM) was first introduced by Birbil and Fang [1]. It simulates the attraction-repulsion mechanism in the electromagnetism theory [2] to solve unconstrained nonlinear optimization problems in the continuous space. This method has been tested and verified in order to prove the efficiency of the EM. It can be converged rapidly, in terms of the number of function evaluations, to the global optimum for the given continuous problems with varying degree of difficulty [1, 3].

Despite its proven capability of solving various optimization problems in the continuous space, EM has been applied to combinatorial optimization problems only in few studies mostly in the field of scheduling problems [4 to 6] and traveling salesman problem [7, 8]. Wu and Chiang [7] and Wu et. al. [8] applied EM to solve the traveling salesman problem (TSP). Their methods, for converting the space of EM from continuous into discrete space, are very simple and inefficient. In other words, they have just rounded off the values. While using the TSP problems in very small sizes (up to 16 cities), those associated results are not satisfactory.

In this paper, we propose a discrete binary version for the electromagnetism-like method (EM) in order to solve combinatorial optimization problems. Our method for converting the continuous version of the original EM [3] into the discrete version is very similar to the method proposed by Kennedy and Eberhart [9]. They converted the continuous version of the particle swarm optimization (PSO) method [10] into the discrete version. This discrete binary version of PSO has been widely used for solving many combinatorial optimization problems like Flow shop scheduling problems [11, 12], traveling salesman problem [13], and polygonal approximation of digital curves [14]. Like the discrete binary version of PSO, our discrete binary version of EM can be used in many combinatorial optimization problems. As an instance, in this paper we use our proposed EM to solve the traveling salesman problem (TSP).

To show the efficiency of our proposed EM on solving the TSP problem we carry out a number of experiments and then we compare our computational results with those taken from the literature [7, 8]. Finally we conclude that our algorithm is more capable of solving the TSP in large sizes.

2 Electromagnetism-Like (EM) Mechanism

The electromagnetism-like method (EM) was first proposed by Birbil and Fang [1] and then revised by Birbil et al. [3] to ensure the convergence of this method. After applying some modifications in the original algorithm, it has been proven that this new revision exhibits global convergence with probability one. The EM applies on optimization problems with continuous variables in the following form, as given in Eqs. (1) and (2).

$$\begin{aligned} \min f(x) \\ \text{s.t. } x \in [L, U] \end{aligned} \quad (1)$$

$$[L, U] := \{x \in R^n \mid L_k \leq x_k \leq U_k, k= 1, 2, \dots, n\} \quad (2)$$

This method uses an attraction-repulsion mechanism to move sample points toward optimality. It is similar to the electromagnetism theory [2] for charged particles. Each sample point (solution) in the solution space is assumed as a charged particle where the charge of a point relates to the objective function of that point. Points with better objective function have more charge than other points. Then, points with more charge (i.e., solutions with better objective function) attract other points, and points with less charge repulse other points. Finally, a total force vector exerted to a point is calculated by adding these attraction-repulsion force vectors and the point is moved to direction of the total force. Table 1 shows the general scheme of the EM and its description of each step.

- **Initialize (Line 1):** In this step, m sample points are selected randomly from the feasible region, which is n dimensional hyper-cube. Then, the objective function value (OFV) of each sample point is computed. This step ends with m points identified, and the point with the best OFV is stored in variable x^{best} .

Table 1. Structure of the EM

ALGORITHM EM(m , MAXITER, LSITER, δ)	
m :	number of sample points
MAXITER:	maximum number of iterations
LSITER:	maximum number of local search iterations
δ :	local search parameter, $\delta \in [0, 1]$
1:	Initialize()
2:	iteration $\leftarrow -1$
3:	while iteration < MAXITER do
4:	Local(LSITER, δ)
5:	$\mathbf{F} \leftarrow \text{CalcF}()$
6:	Move(\mathbf{F})
7:	iteration \leftarrow iteration + 1
8:	end while

- **Local search (Line 4):** This step is used to move sample points toward the local optima, which are close to these points pushed toward the local hills by using a neighborhood search procedure.
- **Calculate force (Line 5):** In this step, a charged-like value is assigned to each point (q^i). The charge of a point is computed according to the efficiency of the OFV of a point (points with better objective function have more charge than others). The charges are computed by Eq.(3).

$$q^i = \exp \left[-n \times \frac{f(x^i) - f(x^{best})}{\sum_{k=1}^m [f(x^k) - f(x^{best})]} \right], \quad i = 1, 2, \dots, m \quad (3)$$

Then, the force between two points is calculated by using a mechanism that is similar to the electromagnetism theory [2] for the charged particles. In this mechanism, the force that is exerted on a point via other points is inversely proportional to the distance among points, and directly proportional to the product of their charges. A point with a better OFV (i.e., larger q^i) attracts other points, and a point with the worse OFV repels the others. At the end of this step, the vector of the total force exerted on each point from other points is calculated. This vector determines the direction of movement for corresponding point in the next step (Line 6 of the EM).

In the revised EM proposed by Birbil et al. [3] needs some modifications in this step of the algorithm. To preclude the premature convergence, Birbil et al. [3] selected one of the points in the population other than the current best point as the "perturbed point", and they modified the method of calculating the force that was exerted to the perturbed point. After applying these modification in this step of the algorithm, they proved that the new revised algorithm exhibits global convergence with probability one. We also use these modifications in our study.

- **Move points along the total force vector (Line 6):** In this step, points are moved along the total force vector that is calculated in the previous step. The length of the movement along the total force vector is selected randomly.

3 Proposed Discrete Binary Version of the EM

According to the above discussion, the EM is restricted to real numbers (i.e., continuous space). However, many optimization problems are set in a space featuring discrete or qualitative distinctions between variables. To meet this requirement, we develop a discrete version of the EM by using an approach that is very similar to Kennedy and Eberhart [6] in order to make a discrete version for a PSO method. The discrete EM essentially differs from the original (or continuous) EM in two characteristics: 1) each point (solution) is composed of the binary variable; and 2) the total force vector computed in Line 5 of the EM shows the chance of the binary variable taking the value one in the next step of the algorithm (Line 6).

To better understand our discrete EM, suppose that $X^i = (x_1^i, x_2^i, \dots, x_n^i)$, $x_n^i \in \{0,1\}$ is point (solution) i in the space and $F^i = (f_1^i, f_2^i, \dots, f_n^i)$ is the total force vector exerted to point i (this vector is calculated in Line 5 of the algorithm). In line 6 of the EM, we select the greatest positive component of F^i . This selected component of F^i has a corresponding component in X^i . Now if this corresponding component of X^i is equal to zero, we change it to 1. Table 2 shows a proposed algorithm for **Move(F)** of our proposed EM.

Table 2. Proposed algorithm for Line 6 of the EM

ALGORITHM Move(F)	
F	: Total force vector calculated in line 5 of the EM
Pop	: Total number of Population (number of solutions)
n	: Number of variables in each solution
1.	While $i < Pop$ do
2:	$\max f \leftarrow 1$
3:	while $j < n$ do
4:	If $f_j^i > \max f$ then
5:	$\max f \leftarrow f_j$
6:	end If
7:	$j \leftarrow j + 1$
8:	end while
9:	$x_{\max f}^i \leftarrow 1$
10:	$i \leftarrow i + 1$
11:	end while

For example, suppose $X^i = (0,1,0,0,0,1)$ consisting. After computing the total force vector according to Line 5 of the EM, the total force vector exerted to solution i is $F^i = (0.2, 1.3, -1.2, 0, 0.7, 2.3, -0.1)$. Now for running command in Line 6 according to the proposed algorithm, we select the largest values of f^i . In this example, the largest positive value of F^i is 2.3 that belongs to f_6^i . So, we change the value of x_6^i from 0 to 1.

At the end of Line 6 of the EM, solution X^i is changed into $X^i = (0,1,0,0,1,1)$. Then, Line 6 is finished and we can go to Line 7 and continue other steps of the algorithm as usual.

4 Using the Proposed EM for Solving the TSP Problem

In this section, we extend the discrete EM to solve the TSP. To do this, our discrete points (solutions) need to be redesigned to represent a sequence of possible tour consisting of m cities, and Line 6 of the EM has to move a point (solution) to a new possible tour. Details are given bellow.

To define points, suppose that point i represents a tour of m cities, and it is shown by $X^i = (x^i_{11}, x^i_{12}, \dots, x^i_{1m}, \dots, x^i_{m1}, x^i_{m2}, x^i_{mm}), x^i_{nk} \in \{0,1\}$. If $x^i_{nk} = 1$, it means that in Step n of tour i we go to city k . Table 3 shows a tour of five cities with the sequence of 1,3,4,2,5. In this figure, we have $x^i_{11} = x^i_{23} = x^i_{34} = x^i_{42} = x^i_{55} = 1$ and other components of X^i are equal to 0. It is worthy noting that in this study all tours start with City 1 remaining the first city in all iterations of the proposed algorithm. Suppose that after executing Line 5 of the proposed EM, the total force vector exerted to the solution is shown in Table4.

Table 3. Values of x^i_{nk} representing a tour as 1,3,4,2,5

		C				
		1	2	3	4	5
Step of the tour	1	1	0	0	0	0
	2	0	0	1	0	0
	3	0	0	0	1	0
	4	0	1	0	0	0
	5	0	0	0	0	1

Table 4. Total force vector exerted to the solution from Table 2 (f^i_{nk})

0	0	0.007	0	0.011
0	-0.001	0	-1.06	0
1.006	0	1.031	0	0.061
0	0.031	0	0	0.703
0	0.005	0	0	1

Now for running Line 6 of the EM, we select the largest positive force in Table 4. The largest force is $f^i_{33} = 1.031$. So, we have to change x^i_{33} from 0 to 1; however, to have a possible solution (i.e., feasible tour) after this change, we also have to apply these changes after changing the value of x^i_{33} as follows:

x_{34}^i should be changed from 1 to 0,
 x_{24}^i should be changed from 0 to 1, and
 x_{32}^i should be changed from 1 to 0.

After applying these changes, solution x^i changes into the form shown in Table 5, representing a tour as 1,4,3,2,5.

Table 5. Values of x_{nk}^i after running line 6 of the EM

1	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	1	0	0	0
0	0	0	0	1

5 Computational Results

To compare our computational results with others, we first review the results reported in the literature [7, 8]. In these studies, each TSP test problem has been solved by EM method 20 times and the best result among these 20 results is reported. The number of iterations of the EM is 1000 and the number of local searches in each iteration is 100 times, in which local searches are applied only to the current best point. The computational results are given in Table 6. In this table, the previous studies cannot solve even very small TSP problems. For example, none of these two studies can solve a problem with 14 cities.

Table 7 shows the computational results of our proposed EM. We have selected all of our test problems from the TSP library [15]. Each test problem has been solved 20 times and the best results as well as average results are reported. It is worthy noting

Table 6. Computational results reported in [7, 8]

Number of cities	Optimal tour	Best result of study [7]	Error of study [7]	Best result of study [8]	Error of study [8]
5	36.9054	36.9054	0.0%	36.9054	0.0%
10- test problem 1	183.1812	183.1812	0.0%	183.1812	0.0%
10- test problem 2	161.6289	161.6289	0.0%	161.6289	0.0%
10- test problem 3	126.7849	126.7849	0.0%	126.7849	0.0%
14	30.879	32.814	6.3%	31.452	1.9%
16	3.2	3.6129	12.9%	3.2	0.0%

Table 7. Computational results of our proposed EM algorithm

Test problem	Number of cities	Number of iterations	Optimum tour	Best result	Error of best result	Average of results
Burma 14	14	150	3323	3323	0%	3323
Gr 17	17	150	2085	2085	0%	2093
Gr 21	21	200	2707	2707	0%	2766
Gr 24	24	200	1272	1272	0%	1328
Beyg 29	29	250	1610	1610	0%	1721
Eli 51	51	400	426	451	5.9%	493

that the number of iterations of our proposed EM is 150 to 500 times (i.e., At least 2 times less than the previous studies). The number of local searches applied to all points in each iteration is 100 times.

To compare the results of Tables 6 and 7, it is clear that our discrete binary version is more capable of solving the TSP by our proposed EM than the previous studies [7, 8]. This conclusion is also clear in Figure 1 and figure 2 which compare the % error related to each algorithm in the given city numbers.

As shown in Table 7, another important conclusion is that our proposed algorithm reports better results by using the less number of iterations comparing to WU and Chiang [7] and Wu et al. [8]. Due to pre-mature convergency, increasing the number of iterations cannot help in improving the quality of results. On the other hand, the lager iterations are not useful for our algorithm because all sample points (except for the perturbed point) are converged to the same solution after a number iterations.

6 Conclusion

We proposed a discrete binary version of the electromagnetism-like method (EM) in order to make this method applicable for combinatorial optimization problems. Then we used our proposed discrete version of EM to solve the traveling salesman problem (TSP). To show the efficiency of our proposed EM on solving the TSP problem, a number of experiments are carried out and the associated results are compared with the results taken from the literature. The quality of solutions is much better than others in terms of the objective function value and the number of iterations.

References

1. Birbil, S.I., Fang, S.C.: An Electromagnetism-like Mechanism for Global Optimization. *Journal of Global Optimization* 25(3), 263–282 (2002)
2. Cowan, E.W.: *Basic Electromagnetism*. Academic Press, New York (1968)

3. Birbil, S.I., Fang, S.C., Sheu, R.L.: On the Convergence of a Population-Based Global Optimization Algorithm. *Journal of Global Optimization* 30, 301–318 (2005)
4. Changa, P.C., Chenb, S.H., Fan, C.Y.: A Hybrid Electromagnetism-like Algorithm for Single Machine Scheduling Problem. *Expert Systems with Applications* (in press)
5. Maenhout, B., Vanhoucke, M.: An Electromagnetic Meta-heuristic for the Nurse Scheduling Problem. *Journal of Heuristics* 13(4), 359–385 (2007)
6. Debelsa, D., Reyckb, B., Leusc, R., Vanhoucke, M.: A Hybrid Scatter Search/ Electromagnetism Meta-Heuristic for Project Scheduling. *European Journal of Operational Research* 169(2), 638–653 (2006)
7. Wu, P., Chiang, H.C.: The Application of Electromagnetism-like Mechanism for Solving the Traveling Salesman Problems. In: *Proceeding of 2005 Chinese Institute of Industrial Engineers Annual Meeting, Taiwan, R.O.C* (2005)
8. Wu, P., Wei, N.C., Fang, H.C.: Revised Electromagnetism-like Mechanism for the Traveling Salesman Problem. In: *Proceeding of the 36th Int. Conf. on Computers and Industrial Engineering, Taipei, Taiwan, pp. 1–9* (2006)
9. Kennedy, J., Eberhart, R.C.: A Discrete Binary Version of the Particle Swarm Algorithm. In: *Proceedings of the World Multi Conference on Systemic, Cybernetics and Informatics, Piscataway, NJ, pp. 4104–4109* (1997)
10. Eberhart, R.C., Kennedy, J.: A New Optimizer Using Particle Swarm Theory. In: *Proceedings of the 6th Int. Symposium on Micro Machine and Human Science, Nagoya, Japan, pp. 39–43* (1995)
11. Liao, C., Tseng, C., Luarn, P.: A Discrete Version of Particle Swarm Optimization for Flowshop Scheduling Problems. *Computers & Operations Research*, pp. 3099–3111 (2007)
12. Tseng, C.T., Liao, C.J.: A Discrete Particle Swarm Optimization for Lot-Streaming Flowshop Scheduling Problem. *European Journal of Operational Research* (2007) doi:10.1016/j.ejor.2007.08.030
13. Li, X., Tian, P., Hua, J., Zhong, N.: A Hybrid Discrete Particle Swarm Optimization for the Traveling Salesman Problem. In: Wang, T.-D., Li, X.-D., Chen, S.-H., Wang, X., Abbass, H.A., Iba, H., Chen, G.-L., Yao, X. (eds.) *SEAL 2006. LNCS, vol. 4247, pp. 181–188*. Springer, Heidelberg (2006)
14. Yin, P.Y.: A Discrete Particle Swarm Algorithm for Optimal Polygonal Approximation of Digital Curves. *Journal of Visual Communication and Image Representation* 15, 241–260 (2004)
15. [http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/\(the TSP Library\)](http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/(the%20TSP%20Library))

A Frequent Pattern Mining Algorithm for Understanding Genetic Algorithms

Minh Nghia Le and Yew Soon Ong

School of Computer Engineering, Nanyang Technological University,
Nanyang Avenue, Singapore 639798
{1emi0005, asysong}@ntu.edu.sg

Abstract. In this paper, we present a Frequent Schemas Analysis (FSA) approach as an instance of Optinformatics for extracting knowledge on the search dynamics of Binary GA using the optimization data generated during the search. The proposed frequent pattern mining algorithm labeled here as *LoFIA* in FSA effectively mines for interesting implicit frequent schemas. Subsequently these schemas may be visualized to provide new insights into the workings of the search algorithm. A case study using the Royal Road problem is used to explain the search performance of Genetic Algorithm (GA) based on FSA in action.

Keywords: Genetic Algorithms; Optinformatics; Frequent Pattern Mining; Schemata.

1 Introduction

Details on how an evolutionary algorithm operates to search on a given landscape is unfortunately still not completely well understood. Knowledge about the search structure of the algorithm may bring new insights to “*how an algorithm search on a problem ?*” and/or “*under what circumstances does an algorithm functions well ?*”. Answers to these questions can be useful to assist one in selecting appropriate solvers for a given problem. In this paper, we particularly narrow our focus to Genetic Algorithm. Several attempts have been made in the past decades to enhance our understandings on the dynamics of GA and its connection to the problem landscape [1], [2]. To reduce the difficulty and complexity of the theoretical model in order to capture the stochastic behaviors of the optimization process, assumptions on infinite population size or use of simple operators are usually made in most studies. However, because of the assumptions made, most approaches cannot be used to analyze the behaviors of the general evolutionary search on a given problem at hand.

As introduced in [3], *optinformatics* refers to *the specialization of informatics for the processing of data generated in optimization so as to extract possibly implicit and potentially useful information and knowledge*. Even though it is generally tough to precisely predict how an evolutionary algorithm would search on a problem, the least one could do is to develop methods in optinformatics that aims to estimate the performance and highlight how the algorithm operates on the problem through the archived optimization data produced during (online) or at the end of the search (offline). In this paper, we present a Frequent Schemas Analysis (FSA) technique for extracting knowledge from

the search process by using the historical optimization data, which are otherwise often discarded. FSA bring about greater understanding of GA dynamics through mining for frequent schemas that exist implicitly within the optimization data. The rest of the paper is organized as follows. Section 2 presents the definition of frequent schema and discusses the hardness of mining frequent schemas on GA data. In Section 3 we present the Frequent Schemas Analysis (FSA) technique for mining and visualizing interesting frequent schemas. Using the proposed FSA, we analyze the premature convergence experience of GA on the Royal Road problem in Section 5 and confirm the conclusion reported in earlier work. Finally, we conclude our findings in Section 6.

2 Frequent Schemas in Genetic Algorithms

Over the past decades, many efforts on the analysis of Genetic Algorithm have been made since the first introduction of schemata and Schema Theorem by Holland. In his work, a schema is defined as a notational device to represent a set of fixed-length genomes (chromosomes) of length L having some alleles in common. It is a string of length L in which position i can take a specific allele value (i.e. 0 or 1 in case of binary string) or $*$ which represents a *don't care* symbol.

Let function $Freq(s, [m, n])$ denote the frequency of schema s in the populations over generations m to n . We define a schema s as *frequent schema* with a level θ in the period $[m, n]$ if and only if $Freq(s, [m, n]) \geq \theta$. One possible interpretation of a frequent schema s is that GA has spent at least θ percentage of its sampling budget on the hyperplane defined by s ; or θ is a lower bound of the probability that a point in the hyperplane s is sampled by GA during the period $[m, n]$. Frequent schemas with a higher value of θ indicate regions with stronger convergence of GA in the period. However, a set of frequent schemas detected with higher value of θ is generally less specific (lower-order schemas) than those of lower values.

Note that each chromosome (binary string) in the data generated by GA in the period of generations can be transformed to a set of items (itemset). Each allele x_i at locus $i \in \Omega = \{0, 1\}$ is converted to an item with index $y_i = |\Omega| \times i + 1 + x_i$. Conversely, locus i and allele x_i can be calculated from y_i as $i = (y_i - 1) / |\Omega|$ and $x_i = (y_i - 1) \bmod |\Omega|$. For example, chromosome 010...1 of length L in an optimization is transformed to a set of items $\{1, 4, 5, \dots, 2L + 2\}$. At this point, the term *frequent schema* and *frequent itemset* are used interchangeable, and the problem of finding frequent schemas on the aggregated GA data over a period can be considered as a problem in Frequent Pattern Mining [4].

Let $\mu_s(t)$ denote the observed average fitness of individuals that belong to schema s . As stated in Holland's book [2], "..., if some schema begins to occupy a large fraction of the population (through consistent above-average performance), its rate of increase will come very close to $[\mu_\xi(t) / \mu(t)] - 1$ ", it is inferred that the frequencies of a schema with consistently *above-average* performance in a period form a *non-decreasing* sequence $\{Freq(s, t)\}_{t=m}^n$ and thus, a subset of consistently above-average schemas will become a part of the set of frequent schemas. With reference to [5], above-average schemas having short and closely spaced fixed positions that are not too unfavorably affected by crossover, termed as *building block*, tend to increase its frequency in the population

as GA progresses, thus reducing the complexity of the problem. The Building Block Hypothesis (BBH) states that GA thus operates on these building blocks, growing them and mixing them with each other in an attempt to find the optimum solution. Since low order building blocks will combine to form a higher order building blocks under BBH and building blocks contribute to the set of frequent schemas as discussed, it is expected that the length of most specific frequent schemas (itemsets) found will increase or *dense* datasets (i.e. data with long itemsets) will be obtained in the later period of the evolution. As discussed in [6], dense dataset often poses a challenge to the algorithms that mine for frequent patterns (or frequent schemas in this context). Thus, Building Block Hypothesis results in this hardness of mining frequent schemas from the dense data generated by GA in the later periods of the evolution.

3 Frequent Schemas Analysis

Instead of focusing on all details that may possibly be irrelevant in each search generation, the idea of Frequent Schemas Analysis (FSA) is to capture relevant information about the schemata evolution for some periods of time through interesting frequent schemas as shown in Figure 1 in order to provide an overview or a sketch of the evolution process across periods.

Optimization data is first collected from the evolution process and divided into consecutive and non-overlapping periods. The sampling of GA in each period of the search space is analyzed by investigating on the set of frequent schemas ($Freq(s, P) \geq \theta$) found in that period. Alternatively, frequent schemas can also be compared across periods to understand the change in GA dynamics. Large value of θ gives more confidence on the located convergence regions but the frequent schemas are generally less specific, thus, interesting information may be not captured.

3.1 Frequent Schemas Mining

Besides requiring a huge amount of memory and computation resources, processing and storing all possible frequent schemas (e.g. at most 2^{30} for 30-bit strings) may not be necessary for the analysis and interpretation purposes. From the possibly numerous frequent schemas, it is up to the analyzer to select *interesting* schemas from the pool to investigate. In this paper, the *interestingness* metric is defined as the longest frequent schema/ itemset (LFS/ LFI in Figure 1) for that longest frequent schemas may provide a

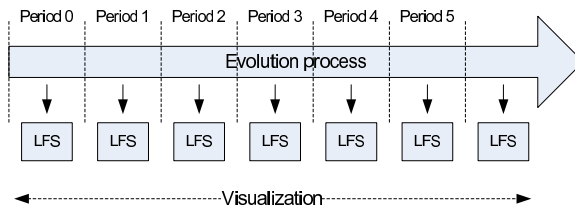


Fig. 1. Frequent Schemas Analysis

sketch on how GA progressively reduces the number of dimensions of its search space or biases its search towards promising convergence regions.

For this purpose, simply applying available algorithms in Frequent Pattern Mining may be very inefficient on some optimization data, such as those generated by the Royal Road problem which will be illustrated in Section 4. Based on the compact data structure *FP-tree* in *FP-growth* algorithm [6], we propose the *LoFIA* algorithm (Algorithm 1) which employs bottom-up and depth-first approach to quickly identify the *longest* frequent schemas from the optimization data. Support (frequency) counting for all frequent itemsets with prefix s is computed through the compact data structure *FP-tree*, instead of having to scan through the entire large dataset. *FP-tree* is reported in [6] to achieve good compactness in most of the time, especially in dense datasets such as GA optimization data.

Algorithm 1. Longest Frequent Itemset Mining Algorithm (LoFIA)

```

1: [ $\mathbf{T}$ ] = Build(Data,  $\theta$ ) {Build FP-tree  $\mathbf{T}$ }
2: Set LongestFrequentItemsets (LFI) = null
3: Set  $l_\theta = 0$ 
4: LFI = LoFIA-Mining(null,  $\mathbf{T}$ , LFI,  $l_\theta$ )
5: return LFI

```

A search from a current tree T_s (i.e. *FP-tree* of prefix s) can be interpreted as finding the longest frequent itemset started with the prefix s . A conditional *FP-tree* $T_{s\alpha}$ is recursively generated from T_s which conforms to the depth-first search strategy (line 11, Algorithm 2). In this way, a complete set of frequent itemsets is found by recording a prefix as frequent itemset (line 9, Algorithm 2) and recursively generating the conditional *FP-trees* from the *FP-tree* of that prefix. The reader is referred to Han's paper [6] for the details of generating conditional *FP-tree* T_s and header table H_s .

Algorithm 2. LoFIA-Mining(s , \mathbf{T}_s , LFI, l_θ)

```

1: Build Header Table  $H_s$ 
2: for each item  $\alpha$  in  $H_s$  do
3:   if Order( $\alpha|H_s$ ) +  $|s| \geq l_\theta$  (Criteria 2) then
4:     if  $|s\alpha| > l_\theta$  then
5:       LFI.clear()
6:        $l_\theta = |s\alpha|$  {update to new expected longest length}
7:     end if
8:     if  $|s\alpha| == l_\theta$  then
9:       LFI.insert( $s\alpha$ )
10:    end if
11:    Build conditional FP-tree  $T_{s\alpha}$  from  $T_s$ 
12:    if  $|s\alpha| + D(T_{s\alpha}) \geq l_\theta$  (Criteria 1) then
13:      LoFIA-Mining( $s\alpha$ ,  $T_{s\alpha}$ , LFI,  $l_\theta$ )
14:    end if
15:  end if
16: end for

```

The speed-up in the performance of *LoFIA* is achieved by incorporating the proposed pruning criteria of mining for longest frequent itemsets in order to avoid processing branches which are *already* known to contain only *short* frequent schemas. Furthermore, parameter l_θ is introduced in the algorithm as an expected lower bound of the maximum length of frequent itemsets at level θ (support threshold). It is worth noting

that l_θ can be used to control a more effective pruning by initializing it to proper expected length of longest frequent itemset (> 0). When no prior knowledge about the length of the longest frequent itemset is available, the initial value of l_θ in the algorithm is 0. The pruning criteria used in the algorithm are based on two theorems which can be derived from the depth-first search strategy and the recursive method to construct conditional FP-tree T_s .

Theorem 1: Denote $D(T_s)$ as the depth of the FP-tree T_s (i.e. length of the longest branch), the length of the longest frequent itemset with prefix s (i.e. L_s) is always less than or equal to $|s| + D(T_s)$: $L_s \leq |s| + D(T_s)$

Theorem 2: Denote $Order(\alpha|H_s)$ as the order of item α in the conditional header table H_s , the longest frequent itemset with prefix $s\alpha$ has its length $L_{s\alpha}$ less than or equal to $Order(\alpha|H_s) + |s|$: $L_{s\alpha} \leq Order(\alpha|H_s) + |s|$

By these theorems, two pruning criteria in Algorithm 2 are derived as: (1) it is not necessary to proceed further on the branch of prefix s if $|s| + D(T_s) < l_\theta$ (Theorem 1) and (2) when having the conditional FP-tree T_s of prefix s and an item α of interest, it is not necessary to proceed further on branch of prefix $s\alpha$ if $Order(\alpha|H_s) + |s| < l_\theta$ (Theorem 2). It is worth noting that different upper bounds for the length of the longest frequent itemset of certain prefix are used in Criteria 1 and 2 to reduce the computational cost in the mining process. In processing the branch with prefix $s\alpha$ which can be pruned by using either one of the two criteria, Criteria 2 requires less computational cost than Criteria 1 as its upper bound is based on the available information $Order(\alpha|H_s)$ instead of building the conditional FP-tree $T_{s\alpha}$. However, that Criteria 2 requires less computational cost comes as a trade-off to a tighter bound provided in Criteria 1 which is based on the depth of $T_{s\alpha}$.

3.2 Frequent Schemas Visualization

A method which is introduced to visualize current convergence regions of GA in one period and the differences observed across periods serves to provide hints to the dynamics of GA. Vectors x of length L which represents the set of M most specific frequent schemas in consecutive periods are plotted against the time axis in the final visualization (Figure 3). The value of element x_i for locus i is then calculated by $x_i = \frac{N_1 - N_0}{M}$, where N_1 and N_0 are the number of schemas in the set has value 1 and 0, respectively, at locus i . Vector x can be considered as the combination of convergence regions in the probabilistic manner, in which $|x_i|$ near 1 indicates a clear overlapping of the interesting schemas at locus i ($x_i = 1/ - 1$ means all schemas agree on allele 1/0, respectively). For example, vector x used to represent the list of the longest frequent schemas $\{111 * 00, 1 * 100*, *11000\}$ is $\{\frac{2}{3}, \frac{2}{3}, 1, \frac{-2}{3}, -1, \frac{-2}{3}\}$.

4 Computation Cost and the Effect of Pruning in LoFIA

In this section, the runtime performance (i.e. total execution time) comparison among the naive FP-growth and LoFIA with different pruning criteria on test data was reported to confirm our comment on the hardness of GA optimization data and illustrate the

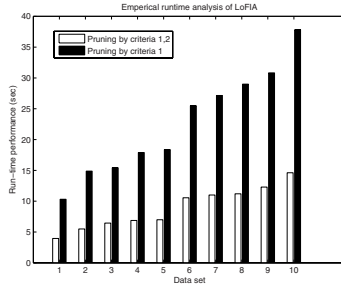


Fig. 2. Runtime analysis of pruning criteria in LoFIA

effect of the pruning criteria in *LoFIA*. Note that the *naive FP-growth* algorithm mines for the longest frequent itemset by filtering the list of frequent itemsets found by *FP-growth* [6]. 10 sets of test data were generated from GA runs on 32-bit Royal Road problem as described in details in Section 5.

For each of the 10 data set, the experiment shows that the *naive FP-growth* could not complete processing the data within the limited runtime of 1 hour which is considerably larger than the runtime performance of our proposed algorithm (< 40 seconds). The results show a significant speed-up of *LoFIA* over the *naive FP-growth* approach. Algorithm 2 using only Criteria 1 was also compared to *LoFIA* as shown in Figure 2. The figure suggests that as the computational cost of building conditional *FP-tree* is diminished by Criteria 2, the combined effect of pruning through both Criteria 1 and 2 in *LoFIA* helps to reduce the overall runtime significantly over the use of Criteria 1 alone.

5 Frequent Schemas Analysis of GA on Royal Road Problem

Royal Road problem is one of the typical test functions for Binary GA, introduced in [7] by Mitchell *et al.* Two common configurations of linear (R1) or non linear reinforcement (R2) for the problem were described in [8]. Under the support of Building Block Hypothesis, the hierarchical structure of building blocks in Royal Road problem was expected to construct a *road* for GA to reach the global optimum. Even though the Royal Road problem was originally designed for GA to perform well, it is worth noting that the Random Mutation Hill Climbing (RMHC) described in [9] actually outperforms GA on the test function. Table 1 shows the average number of evaluations required by each algorithm to reach the optimal solution over 50 independent runs on the problem of R2 type, 32 bits ($K = 4$) and 64 bits ($K = 8$) in which K denotes the

Table 1. Hill-climbing outperforms GA on Royal Road problem

Methods	Royal Road (32K4)	Royal Road (64K8)
GA	7587.32 ± 7045.26	102880.96 ± 71723.45
RMHC	412.22 ± 206.61	5876.86 ± 2595.55

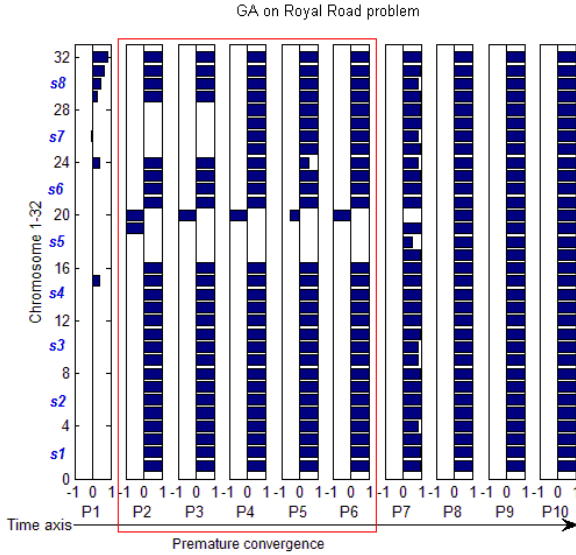


Fig. 3. Frequent schemas analysis of GA at $\theta = 0.8$

size of the fundamental blocks (e.g. s_1 to s_8 in Figure 3) for 32-bit problem, $K = 4$). GA is configured with one-point crossover p_{cross} of 0.8, bit-flip mutation p_{mut} of 0.003, fitness-proportional selection and $popsiz$ e of 50. To investigate what slowed down the GA on the Royal Road problem, Frequent Schemas Analysis (FSA) was used to analyze the archived optimization data of a GA search on the problem of 32 bits. Note that the optimal solution for this problem is $111 \dots 1$ and the maximal fitness is 128. Here, the evolutionary search of 150 generations is divided into 10 periods, with each period consisting of 15 GA search generations. The most specific frequent schemas of each period at $\theta = 0.8$ were then obtained using the proposed *LoFIA* algorithm. The convergence regions of GA (i.e. more than 80% of GA's sampling were on these regions/schemas) in each period are visualized in Figure 3.

Firstly, the plot well illustrates the Building Block Hypothesis in which blocks are shown to be discovered and combined in reducing the complexity of the problem. As expected, the length of longest frequent schemas increases as evolution search progresses. In Figure 3, block s_5 of the frequent schemas was incorrectly identified in Period 2, containing two bits of 0 ($x_{20,19} = -1$). While other blocks of 1's of the convergence regions were correctly discovered by GA in early periods of the evolutionary process, the search had to spend a remarkable amount of time to identify the good configuration for block s_5 (approximately 5 periods P_2 - P_6 or 75 generations) and then were able to converge to the global optimum. It is expected from the observation in Figure 3 that the block with incorrect alleles that hitchhikes in the previous generations requires significant efforts of the search before improvement in the schema is observed thus, suggesting a possible premature convergence of GA on the problem.

Here, the regions or schemas C defined by many correctly discovered blocks of 1 's and one block with more than one bad alleles 0 's (e.g. $C = [11 \dots 1] * 00 * [1 \dots 11]$) are considered as premature convergence regions of GA on the Royal Road landscape. An explanation might go as follows. We define the event of improvement of the search in this case as when an individual with *less* number of incorrect bits (i.e. bit 0) appears in the reproduction pool of the subsequent population. As one-point crossover now operates on the reproduction pool with a large number of individuals satisfying condition C , it becomes unlikely for the operator to bring about an event of improvement in this case. Furthermore, mutating other bits of 1 at the correct blocks of an individual will decrease its fitness significantly due to the loss of building blocks. If the population falls in the premature convergence region with high probability, the mutated offspring with lower fitness will not be selected in the reproduction pool of the next generation. Therefore, it is expected that an event of improvement can only be achieved but at a small probability by mutating bit $0 \rightarrow 1$ while not changing other correct bits. Not to mention that the neutral selection pressure due to the plateau characteristic of the Royal Road fitness landscape defined on region C does not have any reinforcement on the event of improvement, i.e. an improved individual with less number of 0 's can still be lost through selection. Hence, a series of improvement events which are required to identify the correct block (as more than one bad alleles 0 's are presented) is strongly hindered by the combined effect of the above issues.

6 Conclusions

In this paper, a Frequent Schemas Analysis (FSA) technique is introduced as an instance of Optinformatics to provide a comprehensive picture of how the search process evolves, hence bringing new insights into the properties of GA search. In particular, the proposed *LoFIA* mining algorithm is employed in FSA to mine for interesting frequent schemas that exist implicitly within the archived Binary GA data. Through its pruning criteria, the proposed algorithm has shown a significant speed-up in performance over the *naive FP-growth* approach. The schemas of different search periods are then investigated via visualization method. Using the Royal Road problem, we demonstrated the ability of FSA in identifying the premature convergence of GA search which is validated in previous studies [8], [9].

References

1. De Jong, K.: *Evolutionary Computation: A Unified Approach*. MIT Press, Cambridge (2006)
2. Holland, J.: *Adaptation in Natural and Artificial Systems*. MIT Press, Cambridge (1992)
3. Le, M.N., Ong, Y.S., Nguyen, Q.H.: Optinformatics for Schema Analysis of Binary Genetic Algorithms. In: *Genetic and Evolutionary Computation Conference (GECCO) (2008)*
4. Tan, P.-N., Michael Steinbach, V.K.: *Introduction to data mining*. Pearson Addison Wesley, London (2006)
5. Goldberg, D.: *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley, Reading (1989)
6. Han, J., Pei, J., Yin, Y., Mao, R.: Mining Frequent Patterns Without Candidate Generation: A Frequent-pattern Tree Approach. *Data Mining and Knowledge Discovery* 8(1), 53–87 (2004)

7. Mitchell, M., Forrest, S., Holland, J.: The Royal Road for Genetic Algorithms: Fitness Landscapes and GA Performance. In: *Toward a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life*, vol. 1001, p. 48109 (1992)
8. Forrest, S., Mitchell, M.: Relative Building-block Fitness and the Building-block Hypothesis. In: Whitley, L.D. (ed.) *Foundations of Genetic Algorithms 2*, pp. 109–126. Morgan Kaufmann, San Francisco (1993)
9. Mitchell, M., Holland, J., Forrest, S.: When Will a Genetic Algorithm Outperform Hill Climbing. *Advances in Neural Information Processing Systems* 6, 51–58 (1994)

A Generalized Differential Evolution Combined with EDA for Multi-objective Optimization Problems

Wang Chen, Yan-jun Shi, and Hong-fei Teng

School of Mechanical Engineering,
Dalian University of Technology, Dalian, P.R. China 116024
shiyj@dlut.edu.cn

Abstract. This paper proposed a multi-objective evolutionary algorithm (called by GDE-EDA hereinafter). The proposed algorithm combined a generalized differential evolution (DE) with an estimation of distribution algorithm (EDA). This combination can simultaneously use global information of population extracted by EDA and differential information by DE. Thus, GDE-EDA can obtain a better distribution of the solutions by EDA while keeping the fast convergence exhibited by DE. The experimental results of the proposed GDE-EDA algorithm were reported on a suit of widely used test functions, and compared with GDE and NSGA-II in the literature.

Keywords: Generalized differential evolution; estimation of distribution algorithm; multi-objective optimization.

1 Introduction

Many real-world problems in the engineering are multi-objective in nature, e.g., layout optimization of satellite module, mechanical component design, etc. Over the past decades, a large amount of studies had been focused on multi-objective optimization problems (MOOPs) and had obtained a lot of achievement. However, the complexity of some multi-objective optimization problems (e.g., very large search spaces, uncertainty, noises, disjoint Pareto curves, etc.) call for new or alternative approaches.

Differential Evolution (DE) is a stochastic, population-based evolutionary algorithm for global optimization proposed by Kenneth Price and Rainer Storn[1]. DE is capable of converging to the optimal by adopt the distance and direction information from the current population of solutions. Therefore, there have been a lot of the proposed extensions of DE for multi-objective optimization. Chang et al[2] gave the first reported attempt to extend differential evolution for multi-objective problems. Bergery[3] also reported a multi-objective evolutionary algorithm based on differential evolution (called by Pareto Differential Evolution, or PDE) at about the same time as Chang et al. Ref [4] introduced a new version of PDE (called by Self-Adaptive Pareto Differential Evolution, or SPDE).

Besides, there have been many literatures on multi-objective optimization problems with constraints. Iorio and Li[5] firstly proposed the Non-dominated Sorting Differential Evolution (NSDE) with a simple modification of the NSGA-II[6]. Then

they [7] proposed a variation of NSDE. Santana-Quintero and Coello Coello[8] proposed the ϵ -MyDE. Additionally, Ref [9] provided a new generalized differential evolution (GDE) algorithm. To sum up, DE is a very powerful to act as search engine of multi-objective optimization. However, some issues still was raised for multi-objective version of DE. For example, multi-objective DE seemed to have a high convergence rate, but has difficulties to reach the true Pareto front (see for example [8]). The main reason is that DE has no mechanism to extract global information to guide the search for exploring promising areas and to maintain diversity.

To overcome the aforementioned shortcomings, we herein combined Estimation of distribution algorithms (EDA)[10] to spread solutions along the front. EDAs maintained and successively improved a population of candidate solutions and a probability model for promising solutions until some stopping condition was met. Sun et al. [11] had combined DE with EDA to solve some test problems. Zhou et al.[12] proposed a M-MOEA and MOEA can be regarded as a combination of EDA and NSGA-II. The preliminary experimental results show that M-MOEA performs better than NSGA-II. Peter et al.[13] proposed a new algorithm(MIDEA) for evolutionary multi-objective optimization by learning and using probabilistic mixture distributions.

In this study, we combine the DE with EDAs for MOOPs, that is, Generalized Differential Evolution with Estimation Distribution Algorithm (called by GDE-EDA hereinafter). The following sections provide the details of GDE-EDA.

2 The Proposed GDE-EDA Algorithm

Mathematically, constrained multi-objective optimization problem (MOOP) can be presented in the following form without loss of generality:

$$\begin{aligned} \min F(\bar{x}) &= (f_1(\bar{x}), f_2(\bar{x}), \dots, f_k(\bar{x}))^T \\ \text{s.t. } H(\bar{x}) &= (h_1(\bar{x}), h_2(\bar{x}), \dots, h_p(\bar{x}))^T = 0 \\ G(\bar{x}) &= (g_1(\bar{x}), g_2(\bar{x}), \dots, g_m(\bar{x}))^T \leq 0 \end{aligned} \quad (1)$$

where k is the number of objectives to be optimized, p and m is the number of equality constraints and inequality constraint functions respectively. In Eq.1, $\bar{x} = (x_1, x_2, \dots, x_p)$ is decision vector and $F(\bar{x})$ is vector of objective functions. Unlike single-objective problem, there exists a set of Pareto-optimal solutions for $F(\bar{x})$. To provide an alternative solution, we proposed a new algorithm (GDE-EDA) detailed as follows.

2.1 Generalized Differential Evolution

The multi-objective version of DE algorithm used in this study was referenced as Generalized Differential Evolution (GDE)[9]. GDE was an extension of Differential Evolution (DE) for global optimization with an arbitrary number of objectives and constraints. Like DE, GDE also has the mutation, crossover and selection operations. The mutation and crossover operators of GDE are in the same manner as basic DE,

while GDE extends the selection operation of DE for constrained MOOP. In each generation, GDE uses the crossover and mutation operation of DE to generate offspring, and then uses the extension of select operation of DE to form a new population (the next generation).

However, GDE only made improvement on the selection scheme rather than improve the algorithm on the generate scheme of offspring which is the most important operation in EAs. Therefore, we improved this GDE using Estimation of Distribution Algorithm.

2.2 Estimation of Distribution Algorithm

EDA explicitly extract global statistical information from the selected solutions (often called parents) and build a posterior probability distribution model of promising solutions, based on the extracted information. New solutions are sampled from the model thus built and fully or in part replace the old population. To employ EDA, we have to deal with two important issues: (1) how to select the parent solution. We employ the 2-tournament selection for it is a typical selection method used in many applications; (2) how to build a probability distribution model $p(x)$. Since it is impractical to calculate the actual posterior distribution of the promising solutions, most of the existing EDA-like algorithms model the distribution functions by probabilistic graph models or Bayesian networks. We employ Gaussian model with diagonal covariance matrix (GM/DCM)^[14] here because it considers the correlation between random variables, where the joint density function of the G -th generation is written as

$$p_G(x) = \prod_{i=1}^n N(x_i; \mu_i^G, \sigma_i^G), \tag{2}$$

where

$$N(x_i; \mu_i^G, \sigma_i^G) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{1}{2}\left(\frac{x_i - \mu_i}{\sigma_i}\right)^2\right). \tag{3}$$

In Formula (2), the n -dimensional joint probability distribution is factorized as a product of n univariate and independent normal distributions. There are two parameters for each variable required to be estimated in the G -th generation: the mean μ_i^G , and the standard deviation σ_i^G . They can be estimated as

$$\hat{\mu}_i^G = \bar{x}_i^G = \frac{1}{M} \sum_{j=1}^M x_{ji}^G, \tag{4}$$

$$\hat{\sigma}_i^G = \sqrt{\frac{1}{M} \sum_{j=1}^M (x_{ji}^G - \bar{x}_i^G)^2}, \tag{5}$$

where $(x_{1,i}^G, x_{2,i}^G, \dots, x_{M,i}^G)$ are values of the i -th variable of the selected M parent solutions in the G -th generation.

2.3 The GDE-EDA Algorithm

Borrowing from the aforementioned GDE and EDA, we proposed a new multi-objective algorithm, namely GDE-EDA. As mentioned above, GDE used the distance and direction information from the current population to guide its further search, but had no mechanism to extract global information for exploring promising areas and to maintain diversity in solving MOOPs. EDA explicitly extract global statistical information from the selected solutions and can avoid trapping in local optimum. Therefore, we combine GDE with EDAs, attempting to obtain a better algorithm. Figure 1 illuminates the pseudo code of GDE-EDA, where parameter $\delta(CR < \delta \leq 1)$ is used to balance contributions of the global information and the differential information. G , G_{MAX} denote the current generation and max generation of evolution process respectively. $r1, r2, r3$ is random integer.

```

initialize  $G=0, m=0$ 
while  $G < G_{MAX}$ 
  select  $M$  solutions from  $S_G = \{\bar{x}_{1,G}, \bar{x}_{2,G}, \dots, \bar{x}_{N,G}\}$ , construct  $p_G(x) = \prod_{i=1}^n N(x_i; \mu_i^G, \sigma_i^G)$ 
  for each  $\bar{x}_{i,G}, i=1, 2, \dots, N$ 
    for each  $x_{j,G}, j=1, 2, \dots, D$ 
      if  $randb(j) < CR$  or  $j = randr(i)$ 
         $u_{j,G} = x_{j1,G} + F(x_{j2,G} - x_{j3,G})$ 
      else if  $CR \leq randb(j) < \delta$ 
         $u_{j,G}$  is sampled according to  $N(x_i; \mu_i^G, \sigma_i^G)$ 
      else
         $u_{j,G} = x_{j,G}$ 
      end
    end
  end
  selection
   $\bar{x}_{i,G+1} = \begin{cases} \bar{u}_{i,G}, & \text{if } \bar{u}_{i,G} \preceq_c \bar{x}_{i,G} \\ \bar{x}_{i,G}, & \text{otherwise} \end{cases}$ 
  if  $(\forall j: g_j(\bar{x}'_{i,G}) \leq 0)$  or  $(\bar{x}_{i,G+1} == \bar{x}_{i,G})$  or  $(\bar{x}_{i,G} \not\prec_c \bar{x}'_{i,G})$ 
     $m = m + 1$ 
     $\bar{x}_{N+m,G+1} = \bar{x}'_{i,G}$ 
  end
end
while  $m > 0$ 
  for each  $\bar{x} \in S_{G+1} = \{\bar{x}_{1,G+1}, \bar{x}_{2,G+1}, \dots, \bar{x}_{N+m,G+1}\}$ 
    if  $(\forall i: \bar{x} \not\prec_c \bar{x}_{i,G+1})$  or  $(\forall (\bar{x}_{i,G+1}: \bar{x}_{i,G+1} \not\prec_c \bar{x}) \ CD(\bar{x}) \leq CD(\bar{x}_{i,G+1}))$ 
      remove  $\bar{x}$  from  $S_{G+1}$ 
    end
  end
   $m = m - 1$ 
end
end
end
 $G = G + 1$ 
end

```

Fig. 1. The pseudo code of GDE-EDA

3 Experiments

We employed a set of widely used test functions available from the literature [6, 15, 16] to evaluate our proposed approach. The main objective of experiments was

attempted to better understand the strengths and weaknesses of GDE-EDA for different types of test problems.

We used the following metrics to measure the performance[6]: (1) convergence metric Υ , which measure the convergence property of a algorithm by using distance to PF_{true} which is determined by approximating as a combination of piecewise linear segments with the average of these distances defining the metric value. (2) diversity metric Δ [6], which measures distributed extent of a obtained Pareto-optimal solutions over a non-dominated region. The smaller the value Δ , the better the spread extent, and $\Delta=0$ stand for ideal solution set. The diversity metric was formulated as

$$\Delta = \frac{d_f + d_i + \sum_{i=1}^{N-1} |d_i - \bar{d}|}{d_f + d_i + (N-1)\bar{d}}, \quad (6)$$

where the parameter d_f and d_i are the Euclidean distance between the extreme solutions and the boundary solutions of the obtained non-dominated set. The parameter \bar{d} is the average of all distances $d_i, i=1,2,\dots,N-1$, assuming that there are N Pareto-optimal solutions.

The experimental results are compared with that of the state-of-the-art in the area, NSGA-II and GDE.

The size of the population of NSGA-II, GDE and GDE-EDA was 100 and all approaches are run for a maximum of 25000 function evaluations. We use the simulated binary crossover (SBX) operator and polynomial mutation for NSGA-II. Suitable control parameter values of three algorithms for each problem were found by trying out a list of different control parameter values.

3.1 Bi-objective Test Problems

Firstly, GDE-EDA was used to solve the widely used bi-objective optimization problem[6]. All these problems have two objective functions and none of these problems have any constraint. Control parameters for three algorithms are shown in Table 1.

Table 1. Control parameters for the three algorithms

problem	NSGA-II				GDE		GDE-EDA		
	p_c	p_m	η_c	η_m	CR	F	CR	δ	F
Schaffer	0.9	$0.1/n$	20	20	0.5	0.3	0.1	0.2	0.4
ZDT1	0.9	$0.1/n$	20	20	0.3	0.5	0.1	0.15	0.5
ZDT3	0.9	$0.1/n$	20	20	0.2	0.2	0.17	0.2	0.2
ZDT6	0.9	$0.1/n$	20	20	0.2	0.2	0.17	0.2	0.2

3.2 Constrained Bi-objective Test Problems

Then, some constrained bi-objective problems were used to test the constraint-handling ability of GDE-EDA. All these problems^[15] have two objective functions and two constraint functions. Control parameters for three algorithms are shown in Table 2.

Table 2. Control parameters for the three algorithms

problem	NSGA-II				GDE		GDE-EDA		
	P_c	P_m	η_c	η_m	CR	F	CR	δ	F
CONSTR	0.9	$0.1/n$	20	20	0.8	0.5	0.7	0.8	0.5
SRINIVAS	0.9	$0.1/n$	20	20	0.8	0.5	0.7	0.85	0.8
TANAKA	0.9	$0.1/n$	20	20	0.2	0.2	0.7	0.85	0.8

3.3 Tri-objective Test Problems

GDE-EDA was also used to solve the widely used tri-objective test problems, i.e., DTLZ1, DTLZ4 and DTLZ7[16], and compared with GDE3 and NSGA-II. All these problems have three objectives and none of these problems have any constraint. Control parameters for three algorithms are shown in Table 3.

Table 3. Control parameters for the three algorithms

problem	NSGA-II				GDE		GDE-EDA		
	P_c	P_m	η_c	η_m	CR	F	CR	δ	F
DTLZ1	0.9	$0.1/n$	20	20	0.1	0.5	0.1	0.2	0.5
DTLZ4	0.9	$0.1/n$	20	20	0.1	0.3	0.1	0.2	0.4
DTLZ7	0.9	$0.1/n$	20	20	0.2	0.3	0.1	0.15	0.2

3.4 Results and Discussion

We made experiments on the aforementioned test functions, including bi-objective with constrain or not, tri-Objective test Problems. The experimental results of GDE-EDA were compared with that of GDE3 and that of EDA, and shown on Table 4 and Table 5 using two metrics measure the performance for comparative study. Table 4 showed the mean and variance of the convergence metric γ obtained from NSGA-II, GDE and GDE-EDA. We can see that on most of problems, GDE-EDA is able to find better convergence solutions than other two algorithms on metric γ . These results indicated that GDE-EDA kept the fast convergence exhibited by DE. Table 5 showed the mean and variance of the diversity metric Δ obtained from all three approaches. Considering the results from the metric Δ , we can believe that GDE-EDA was able to find better distribution solutions than other two algorithms except in ZDT3 problem.

Table 4. Statistic results of the convergence metric γ

	NSGA-II		GDE		GDE-EDA	
	mean	variance	mean	variance	mean	variance
SCH	0.008022	0.000022	0.007966	0.000029	0.006521	0.000021
ZDT1	0.001816	0.000004	0.000344	0.000000	0.000261	0.000008
ZDT3	0.012311	0.000384	0.010493	0.000343	0.009781	0.000327
ZDT6	0.008695	0.000002	0.003626	0.000002	0.003122	0.000000
CONSTR	0.052367	0.085623	0.049658	0.005297	0.047253	0.000981
SRINIVAS	0.025254	0.000421	0.021833	0.000562	0.019237	0.000346
TANAKA	0.026427	0.000402	0.022866	0.000475	0.020895	0.000340
DTLZ1	0.008583	0.000084	0.007678	0.000063	0.007273	0.000062
DTLZ4	0.381828	0.044978	0.338800	0.134357	0.262161	0.038408
DTLZ7	0.003537	0.000006	0.002978	0.000004	0.002581	0.000004

Table 5. Statistic results of the diversity metric Δ

	NSGA-II		GDE		GDE-EDA	
	Mean	variance	mean	variance	mean	variance
SCH	0.466607	0.001654	0.399495	0.001301	0.333344	0.000934
ZDT1	0.407005	0.000065	0.342632	0.000043	0.297462	0.000035
ZDT3	0.555183	0.000753	0.507527	0.000736	0.495982	0.000747
ZDT6	0.437205	0.000044	0.361424	0.000035	0.301597	0.000027
CONSTR	0.794079	0.005043	0.633901	0.004201	0.568596	0.003269
SRINIVAS	0.405889	3.137228	0.318545	1.798468	0.316344	1.651079
TANAKA	0.795027	0.001342	0.748506	0.006955	0.692994	0.001208
DTLZ1	0.806938	0.005576	0.627404	0.003728	0.596731	0.003283
DTLZ4	0.698626	0.101996	0.638099	0.106633	0.597793	0.097992
DTLZ7	0.677810	0.297248	0.628380	0.309496	0.597150	0.294422

Therefore, it seemed that GDE-EDA outperformed GDE3 and EDA for most of experiments on two metrics measure from the results obtained so far. The reason may be that GDE-EDA combined the advantages of GDE and EDA, which can extract the global information of population for exploration and use the differential information by DE for exploitation, and control two parameters to balance the trade-off between exploration and exploitation.

4 Conclusion and Future Research

In this study we proposed a multi-objective evolutionary algorithm (GDE-EDA) to tackle both unconstrained and constrained multi-objective problems. The proposed GDE-EDA algorithm employed EDA to extract the global information of search for producing better distribution solutions, and employed DE to obtain the fast convergence. The experimental results indicated that GDE-EDA can generate Pareto-optimal solution set with better convergence and better distribution than EDA and GDE3. Further studies on employing different variations of EDA and DE for better performance will be summarized in our next work.

Acknowledgements

This work was supported by National High-tech R&D program of P R China (Grant No. 2006AA04Z109), National Natural Science Foundation of P R China (Grant Nos. 50575031) and National Defense Basic Scientific Research Project of P R China (Grant No. B0920060901).

References

1. Storn, R., Price, K.: Differential Evolution—A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization* 11(4), 341–359 (1997)
2. Chang, C.S., Xu, D.Y., Quek, H.B.: Pareto-Optimal Set Based Multiobjective Tuning of Fuzzy Automatictrain Operation for Mass Transit System. *Electric Power Applications, IEE Proceedings* 146(5), 577–583 (1999)

3. Bergey, P.K.: An Agent Enhanced Intelligent Spreadsheet Solver for Multi-Criteria Decision Making. In: Proceedings of the Fifth Americas Conference on Information Systems, Milwaukee, Wisconsin, USA, pp. 966–968 (1999)
4. Abbass, H.A.: The Self-Adaptive Pareto Differential Evolution Algorithm. In: Proceedings of the 2002 Congress on Evolutionary Computation, CEC 2002 (2002)
5. Iorio, A.W., Li, X.: Solving Rotated Multi-Objective Optimization Problems Using Differential Evolution. In: Webb, G.I., Yu, X. (eds.) AI 2004. LNCS (LNAI), vol. 3339, pp. 861–872. Springer, Heidelberg (2004)
6. Deb, K., et al.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6(2), 182–197 (2002)
7. Iorio, A.W., Li, X.: Incorporating Directional Information within a Differential Evolution Algorithm for Multi-Objective Optimization. In: Proceedings of the 8th annual conference on Genetic and evolutionary computation (2006)
8. Santana-Quintero, L.V., Coello, C.A.C.: An Algorithm Based on Differential Evolution for Multi-Objective Problems. *International Journal of Computational Intelligence Research* 1(2), 151–169 (2005)
9. Kukkonen, S., Lampinen, J.: GDE3: The Third Evolution Step of Generalized Differential Evolution. In: 2005 IEEE Congress on Evolutionary Computation (2005)
10. Mühlenbein, H., Paaß, G.: From Recombination of Genes to the Estimation of Distribution Algorithms I. LNCS, vol. 1411, pp. 178–187 (1996)
11. Sun, J., Zhang, Q., Tsang, E.P.K.: DE/EDA: A New Evolutionary Algorithm for Global Optimization. *Information Sciences* 169(3–4), 249–262 (2005)
12. Zhou, A., Zhang, Q., Jin, Y., et al.: A Model-based Evolutionary Algorithm for Bi-Objective Optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation, pp. 2568–2575 (2005)
13. Bosman, P.A.N., Thierens, D.: Multi-Objective Optimization with Diversity Preserving Mixture-Based Iterated Density Estimation Evolutionary Algorithms. *International Journal of Approximate Reasoning* 31(3), 259–289 (2002)
14. Larrañaga, P., Lozano, J.A.: Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation. Kluwer Academic Publishers, Dordrecht (2002)
15. Nebro, A.J., et al.: A Cellular Genetic Algorithm for Multiobjective Optimization. *NICSO*, pp. 25–36 (2006)
16. Deb, K., et al.: Scalable Test Problems for Evolutionary Multi-Objective Optimization. *Evolutionary Multiobjective Optimization*, pp. 105–145 (2005)

A New Hybrid Ant Colony Optimization Algorithm for the Traveling Salesman Problem

Xiaoxia Zhang and Lixin Tang

The Logistics Institute, Northeastern University, Shenyang, China
aszhangxx@163.com, qhjytlx@mail.neu.edu.cn

Abstract. This paper presents a novel hybrid ant colony optimization approach (ACO&PR) to solve the traveling salesman problem (TSP). The main feature of this hybrid algorithm is to hybridize the solution construction mechanism of the ACO with path relinking (PR), an evolutionary method, which introduces progressively attributes of the guiding solution into the initial solution to obtain the high quality solution as quickly as possible. Moreover, the hybrid algorithm considers both solution diversification and solution quality, and it adopts the dynamic updating strategy of the reference set and the criterion function restricting the frequencies of using the path-relinking procedure to accelerate the convergence towards high-quality regions of the search space. Finally, the experimental results for benchmark TSP instances have shown that our proposed method is very efficient and competitive to solve the traveling salesman problem compared with the best existing methods in terms of solution quality.

Keywords: Ant colony optimization; traveling salesman problem; path relinking.

1 Introduction

The traveling salesman problem (TSP) [1] has been an important problem in the field of distribution and logistics. The traveling salesman problem (TSP) consists in sequentially visiting a set of cities only once and finally returning to the original city of departure. The objective is to determine a minimum distance of a tour passing through each city once and only once. The TSP is clearly NP-hard combinatorial optimization problem and difficult to solve. There have been important advances in the development of exact and approximate algorithms. Exact solution methods can only be used for very small instances, so for real-world problems, researchers have to rely on and resort to approximate or heuristic methods in solving the problem.

Most published research for the TSP has concentrated on the development of heuristics. The majority of the research has emphasized particularly on designing efficient and effective heuristics. Some heuristic approaches have been proved to be very effective both in terms of execution times and quality of the solutions achieved. For example, Glover [2] has devised a very efficient KL (Lin–Kernighan) heuristic which uses 2-Opt and 3-Opt moves to improve his solution within a tabu search algorithmic framework. The KL heuristic can obtain a high-quality solution for instances with

hundreds of cities in short time. Most recent research has focused on employing the advanced meta-heuristic techniques such as genetic algorithms [3], simulated annealing [4] and tabu search [5]. These meta-heuristic algorithms can obtain a high quality of solutions on small instances, but they perform quite poorly on large TSP instances and usually consume substantial computing times and several parameter settings [6].

Ant colony optimization (ACO) is a new meta-heuristic and a very effective local search algorithm. The first ACO algorithm was developed by Dorigo, Maniezzo and Colomi [7] in 1991 and successfully applied to the traveling salesman problem (TSP) based on the path-finding abilities of real ants [8,9]. Chu et al. [10] have designed the parallel formulation for the ant colony system and several communication strategies were used to update the pheromone levels. Nevertheless, the experiments were only tested on several instances. Because the traveling salesman problem is very complicated, the basic algorithms can't satisfy different information needs. Many researchers have proposed new methods to improve the original ACO and applied them successfully to a whole range of different problems, such as the quadratic assignment problem [11], vehicle routing problem [12], etc. It is a trend to combine ACO with other algorithms to solve very large scale of the traveling salesman problem.

The distinctive features of this paper differs from all the papers described above on following points: First, the main idea of this paper is to hybridize the solution construction mechanism of the ACO with path relinking (PR), an evolutionary method, which results in a novel approach that we call ACO&PR. Within the ACO procedure, we embed PR method into the ACO procedure to improve the solutions. Second, despite it takes much longer time to create improved solutions by exploring trajectories that connect high-quality solutions, the dynamic updating strategy of the reference set is adopted to accelerate the convergence towards high-quality regions of the search space. Finally, the experimental results have shown that the ACO&PR algorithm is to be very efficient and competitive in terms of solution quality.

The paper is organized as follows. In Section 2, we describe the basic of ACO algorithm for the traveling salesman problem. A brief review of path relinking and the design of its algorithm are given in Section 3. In Section 4, we present ACO&PR algorithm to tackle the traveling salesman problem. The computational results are reported in Section 5. Finally, the conclusion is given in Section 6.

2 Ant Colony Optimization for TSP

Ant colony optimization (ACO) [9] simulates the behavior of ant colonies in nature as they forage for food and find the most efficient routes from their nests to food sources. The ACO method includes the two basic steps of construction of solutions and pheromone updating. In the TSP ant colony optimization, the artificial ants simulate tours and successively select customers to visit, until all customers have been visited. To select the next customer j , the ant k at the current position of customer i uses the following probabilistic formula:

$$j = \begin{cases} \arg \max_{l \in M_k} [\tau_{il} \cdot [\eta_{il}]^{\beta}], & \text{if } q \leq q_0 \\ S, & \text{otherwise} \end{cases} \quad (1)$$

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij}[\eta_{ij}]^\beta}{\sum_{l \in M_k} \tau_{il}[\eta_{il}]^\beta} & , j \in M_k \\ 0 & , \text{otherwise} \end{cases} \quad (2)$$

where τ_{il} is equal to the amount of pheromone trail, β is a parameter and weighs the importance of distance. The value q is a random uniform variable in $[0,1]$, and the value q_0 ($0 \leq q_0 \leq 1$) is a parameter which determines the relative importance of exploitation versus exploration. If $q \leq q_0$ then the best edge, depending on Equation (1), is selected; otherwise an edge is selected according to S . S represents a random variable selected according to the probabilistic distribution as in Equation (2). p_{ij}^k is the probability with which ant k chooses to move from customer i to customer j . The visibility η_{il} is defined as the inverse of the arc length, and locations already visited by an ant are stored in the ants working memory M_k .

The pheromone trail is updated both locally and globally. Local updating is performed during solutions construction while global updating is performed at the end of the constructive phase. We adopt local and global updating rules presented in [9]. In original ACO, only the best solution is used to globally modify the pheromone trail.

3 Path Relinking for TSP

Path-relinking is an enhancement to the basic ACO procedure, leading to significant improvements in solution quality. Path relinking (PR) [13] is a novel evolutionary method, and it was introduced by Glover and Laguna [14] for integrating diversification and intensification in tabu search. This approach generates new solutions by exploring trajectories that connect high quality solutions in a reference set *RefSet*. The reference set *RefSet* is a set of feasible solutions, and it consists of a total of b_1 high quality solutions (*RefSet*₁) and b_2 diverse solutions (*RefSet*₂), so we use two different methods to generate the feasible solutions. The first one is ACO algorithm, and the second one is a greedy heuristic. The greedy heuristic is similar to the nearest neighbor heuristic for determining the sequence of a tour is near at hand. Each trajectory starts from one of the solutions, called an initial solution, and leads toward the other solution, called a guiding solution. This trajectory is accomplished by performing moves on the current solution, such that attributes in the guiding solution are gradually introduced, and attributes not in the guiding solution are increasingly dropped. Identical parts of two solutions should remain unchanged during the process. Unlike genetic algorithms, where two solutions are randomly chosen from the population to generate new solutions, path relinking utilizes systematic, deterministic rules for combining solutions [15]. During the path relinking phase, a series of intermediate solutions are generated by moving away from the initiating solution. These solutions contain fewer attributes from the initiating solution and more from the guiding solution as the current solution moves along the trajectory. The relinked path (trajectory) may encounter intermediate solutions that may be better than the initiating or guiding solution. For instance, in Fig. 1, the path from A to B (solid line) does not

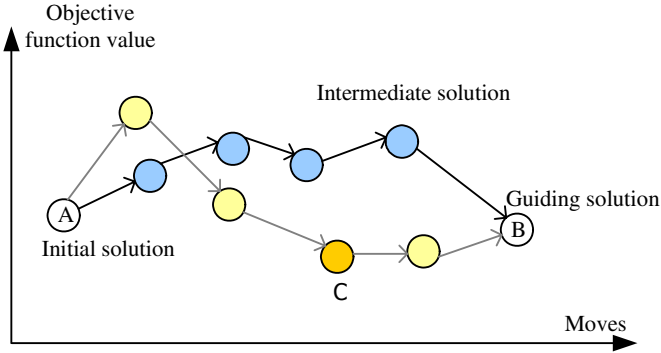


Fig. 1. Process flow of path relinking

yield solutions improving A or B, while the path from A to B (dashed line) can produce two better solutions.

4 ACO&PR Algorithm

In this section we describe the ACO&PR algorithm, which is to hybridize the solution construction mechanism of ACO and path relinking (PR) to solve the TSP. Meanwhile, path relinking procedure is embedded into the ACO procedure to improve solutions. The method has both the advantages of ant colony optimization, the ability to find the higher performance solutions, and that of path relinking, the ability to explore different parts of the solution space and to find better solutions. Our proposed hybrid algorithm consists of the following sections.

4.1 The Improvement Method

After new solutions generated, we attempt to improve the solution quality by applying improvement strategy. 2-opt heuristic is applied at all times to act as an improvement heuristic. Path-relinking procedure is embedded into ACO process and is executed to improve the solutions. To save computation time, it isn't necessary to call the path-relinking procedure at every solution or iteration. We use a criterion function to restrict the frequencies of using the path-relinking procedure, that is, the path-relinking procedure can be executed only if higher quality solutions are obtained. Let δ^{new} be a new solution sequence, δ^* be the best solution sequence. $f(\delta^{new})$ and $f(\delta^*)$ represent the objective function values of the solutions δ^{new} and δ^* , respectively. The criterion function is defined as following:

$$\varepsilon = \varepsilon_0 (f(\delta^{new}) - f(\delta^*)) / f(\delta^*) \quad (3)$$

Let $\varepsilon^* = \varepsilon_0^* (1 - CN / CN_{max})$, ε_0 and ε_0^* are parameters ($\varepsilon_0 = 0.9 \sim 1.0$, $\varepsilon_0^* = 0.2$). CN and CN_{max} represent the current iteration number and the maximum iterations, respectively. A new solution is improved by the path-relinking procedure if its

value $\varepsilon \leq \varepsilon^*$. It is unnecessary to call the path-relinking procedure to improve the new solution if its value $\varepsilon > \varepsilon^*$. The small ε^* has, the less solutions need to be improved using the path-relinking procedure. The method of the criterion function can reduce the computational requirements to reasonable levels.

4.2 A Reference Set Update Method

The quality and level of diversity of the solutions included in the reference set *RefSet* have a major impact on the quality of the generated solutions. In our method, *RefSet* is updated by considering the different solutions obtained using path relinking and the general ACO method. The function $dist(\delta_j, \delta')$ is defined as the degree of diversification between the two solutions δ_j and δ' . The $dist(\delta_j, \delta')$ can be obtained by computing the number of the different arcs between the two solutions. The bigger the values of the $dist(\delta_j, \delta')$ is, the more diverse two solutions are. For each solution δ_j in the set *NewSet* of new solutions obtained using the path relinking method, the maximum $dist_{max}(\delta_j)$ to the reference set is computed according to the following relationship:

$$dist_{max}(\delta_j) = \max_{\delta' \in RefSet} \{dist(\delta_j, \delta')\}, \quad \delta_j \in NewSet \quad (4)$$

The reference set is dynamically updated based on the quality and diversity of the new solutions. To become an element of *RefSet*₁, a new solution should have a better objective function value than the solution with the worst objective value in *RefSet*₁. A new solution may become an element of *RefSet*₂ only if the new solution has a bigger $dist_{max}(\delta_j)$ value than the solution with the smallest $dist_{max}(\delta_j)$ value in *RefSet*₂. Compute the objective function values in the *RefSet*₁, and sort the solutions by increasing their objective function values, where the best solution is the first one in the list.

The terminating criterion is the maximum number of iterations. The procedure is repeated until this number is achieved. The hybrid path relinking procedure steps should be summarized as follows:

- Step 1: Generate initial solutions. Initialize data and generate different initial solutions with ACO algorithm and the greedy heuristic.
- Step 2: Build the reference set (*RefSet*). Select b_1 high quality solutions and b_2 diverse solutions from initial feasible solutions, and then put $RefSet = b_1 + b_2$ solutions in the reference set.
- Step 3: Construct a feasible solution with ACO algorithm.
- Step 4: Improvement solutions. After the feasible solution has been generated, the 2-opt heuristic is always applied to improve solutions. Compute the criterion function value ε of the solution and parameter value ε^* respectively. If its value $\varepsilon \leq \varepsilon^*$, the path-relinking procedure is executed to improve the solution; otherwise, go to Step 7.

- Step 5: Compute the objective function values of the solution and find the best solution $BestSol$ in the reference set.
- Step 6: Update the reference set and the best solution $Bestsol$. Compute $f(\delta_j)$ and $dist_{max}(\delta_j)$, replace the solution of the worst objective value in $RefSet_1$ with the high quality, or the solution of the smallest $dist_{max}(\delta_j)$ value in $RefSet_2$ with a bigger $dist_{max}(\delta_j)$ value. Search for the best solution in new reference set. The best solution $Bestsol$ is updated if the improvement generates a better solution than $BestSol$.
- Step 7: Repeat Steps 4-6 until a terminating criterion is met.

5 Computational Results

In this section, we present computational results of our proposed algorithm, which was coded in the visual C++ and executed on an IBM computer with 512MB RAM and 1600Mhz CPU Speed. To evaluate validity of our proposed algorithm for the traveling salesman problem, the performance of our algorithms was tested on benchmark problems and can be downloaded from the OR Library at the website with <http://www.iwr.uni-heidelberg.de/iwr/Comopt/soft/TSPLIB95/TSPLIB.html>. Many parameters exist in the ACO&PR algorithm, and the values of these affect directly or indirectly the final solution quality. For these test problems, the best known solution for eil51 has been proven optimal. Therefore, most of the parameter values have been determined on the eil51 problem by the numerical experiments. To better understand how β parameter affects the efficiency of the solution process. Different β values were tested in simulation. Simulation results reveal that the parameter $\beta \in \{3,4,5,6\}$ converges faster than with other values. That is, a better solution can be obtained in the same number of or fewer iterations. Experiments have been done with the following parameter settings: $q_0 \in [0.65,0.90]$, $\beta \in \{3,4,5,6\}$, $\rho \in [0.04,0.06]$, $|RefSet|=10$ and $\tau_0 = 0.01$. All the tests have been carried out for maximum iterations $CN_{max}=2000$.

We compare ACO&PR with a number of the better methods available for the TSP, and the results of some problems are described in Table 1, where KniesG refers to Kohonen Network Incorporating Explicit Statistics Global [16], KniesL to a Local version by Aras [16], SA to Simulated Annealing by Budinich [17], Budinich to Budinich's SOM [17], ESOM to the expanded SOM by Leung [18], ACO to pure ACO, and ACO&PR is the algorithm we proposed. It has shown that particular construction mechanism of ACO can find the better solutions, but ACO&PR performs consistently better than the general ACO. The ACO&PR algorithm has also shown to be competitive with the best existing methods in terms of solution quality. Moreover, during this experiment, some solutions are as close as the best solution published so far. Table 1 also shows the gap between the best known solution and our hybrid algorithm. The gap is defined as the percentage increase in the objective function value relative to the best known. A zero gap indicates that best known solutions should be obtained. In comparisons to the best known solutions, ACO&PR can produce very good solutions, and the average deviation is only 0.34%. These percentages indicate

Table 1. Comparison of heuristics for the traveling salesman problem

Instances	Optimal solution	Average deviations(%)						
		KniesG	KniesL	SA	Budinich	ESom	ACO	ACO&PR
eil51	426	2.86	2.86	2.33	3.1	2.1	2.11	0
st70	675	2.33	1.51	2.14	1.7	2.09	2.37	0
eil76	538	5.48	4.98	5.54	5.32	3.89	3.15	0
rd100	7910	2.62	2.09	3.26	3.16	1.96	2.45	0
eil101	629	5.63	4.66	5.74	5.24	3.43	2.70	0.32
lin105	14383	1.29	1.98	1.87	1.71	0.25	2.59	1.10
pr107	44303	0.42	0.73	1.54	1.32	1.48	2.45	1.14
pr124	59030	0.49	0.08	1.26	1.62	0.67	1.62	0.08
bier127	118282	3.08	2.76	3.52	3.61	1.70	3.17	1.27
pr136	96772	5.15	4.53	4.90	5.20	4.31	2.61	0.03
pr152	73682	1.29	0.97	2.64	2.04	0.89	2.51	0.13
rat195	2323	11.92	12.24	13.29	11.48	7.13	2.41	0.30

that our proposed method is very efficient and competitive to solve the traveling salesman problem compared with other existing methods in terms of solution quality.

6 Conclusions

In this paper, we propose an ACO&PR algorithm, a hybrid ant colony optimization approach, which hybridizes the solution construction mechanism of ACO with the path relinking. Within the ACO procedure, we embed PR method into the ACO algorithm to improve the solutions. The experimental results have shown that the hybrid algorithm produces optimal or near-optimal solutions on the benchmark instances and it produces uniformly higher performance solutions relative to the other competing heuristics on the traveling salesman problem.

Acknowledgments. This research is partly supported by National Natural Science Foundation for Distinguished Young Scholars of China (Grant No.70728001), National 863 High-Tech Research and Development Program of China through approved No.2006AA04Z174 and National Natural Science Foundation of China (Grant No. 60521003,60674084).

References

1. Siqueira, P.H., Steiner, M.T.A., Scheer, S.: A New Approach to Solve the Traveling Salesman Problem. *Neurocomputing* 70, 1013–1021 (2007)
2. Glover, F.: Tabu Search-part II. *ORSA Journal of Computing* 12(1), 4–32 (1990)
3. Affenzeller, M., Wanger, S.: A Self-adaptive Model for Selective Pressure Handling within the Theory of Genetic Algorithms. In: Moreno-Díaz Jr., R., Pichler, F. (eds.) EUROCAST 2003. LNCS, vol. 2809, pp. 384–393. Springer, Heidelberg (2003)

4. Budinich, M.: A Self-organizing Neural Network for the Traveling Salesman Problem that is Competitive with Simulated Annealing. *Neural Computation* 8, 416–424 (1996)
5. Liu, G., He, Y., Fang, Y., Oiu, Y.: A Novel Adaptive Search Strategy of Intensification and Diversification in tabu search. In: *Proceedings of Neural Networks and Signal Processing*, Nanjing, China (2003)
6. Baraglia, R., Hidalgo, Perego, R.: A Hybrid Heuristic for the Traveling Salesman Problem. *IEEE Transactions on evolutionary computation* 5(6), 613–622 (2001)
7. Colomi, A., Dorigo, M., Maniezzo, V.: Distributed Optimization by Ant Colonies [A]. In: *Proceedings of the First European Conference of Artificial Life (ECAL 1991)*, pp. 134–142. Elsevier, Amsterdam (1991)
8. Dorigo, M., Maniezzo, V., Colomi, A.: Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics* 26(1), 29–41 (1996)
9. Dorigo, M., Gambardella, L.M.: Ant Colonies for the Traveling Salesman Problem. *Bio-Systems*, 4373–4381 (1997)
10. Chu, S.C., Roddick, J.F., Pan, J.S.: Ant Colony System with Communication Strategies. *Information Science* 167(1-4), 63–76 (2004)
11. Maniezzo, V., Colomi, A.: The Ant System Applied to the Quadratic Assignment Problem. *IEEE Transactions on Knowledge and Data Engineering* 11(5), 769–778 (1999)
12. Bullenheimer, B., Hartl, R.F., Strauss, C.: An Improved Ant System Algorithm for the Vehicle Routing Problem. *Annals of Operation Research*, 89319–89328 (1999)
13. Marti, M., Laguna, M., Glover, F.: Principles of Scatter Search. *European Journal of Operational Research* 169, 359–372 (2006)
14. Glover, F., Laguna, M., Marti, M.: Fundamentals of Scatter Search and Path Relinking. *Control and Cybernetics* 39(3), 653–684 (2000)
15. Ho, S.C., Gendreau, M.: Path Relinking for the Vehicle Routing Problem. *Journal of Heuristics* 12, 55–72 (2000)
16. Aras, N., Oommen, B.J., Altinel, I.K.: The Kohonen Network Incorporating Explicit Statistics and its Application to the Traveling Salesman Problem. *Neural Networks* 12(9), 1273–1284 (1999)
17. Budinich, M.: A Self-organizing Neural Network for the Traveling Salesman Problem that is Competitive with Simulated Annealing. *Neural Computation* 8, 416–424 (1996)
18. Leung, K.S., Jin, H.D., Xu, Z.B.: An Expanding Self-organizing Neural Network for the Traveling Salesman Problem. *Neurocomputing* 62, 267–292 (2004)

A Novel PSO-DE-Based Hybrid Algorithm for Global Optimization

Ben Niu* and Li Li*

School of Management, Shenzhen University,
Shenzhen, 518060 P.R. China
drniuben@gmail.com, llii318@163.com

Abstract. This paper presents a new hybrid global optimization algorithm PSODE combining particle swarm optimization (PSO) with differential evolution (DE). PSODE is a type of parallel algorithm, in which PSO and DE are executed in parallel to enhance the population with frequent information sharing. To demonstrate the effectiveness of the proposed algorithm, four benchmark functions are performed, and the performance of the proposed algorithm is compared to PSO and DE to demonstrate its superiority.

Keywords: Particle swarm optimization; differential evolution; global optimization.

1 Introduction

The particle swarm optimization (PSO) is motivated from the simulation of simplified social behavior first developed by Kennedy and Eberhart [1, 2]. Due to its simplicity in coding and consistency in performance, it has already been widely used in many areas [3].

Differential evolution (DE) is a population-based parameter optimization technique originally proposed by Price [4]. In DE new individuals are generated by mutation and DE's crossover, which cunningly uses the variance within the population to guide the choice of new search points. Although DE is very powerful, there is very limited theoretical understanding of how it works and why it performs well.

In recent years, some attempts have been made to combine the merits of PSO and DE in the context of hybrid methods. Zhang WJ and Xie X F. [5] introduced a hybrid PSO with DE, in which the bell-shaped mutations with consensus on the population diversity by DE operator. Hendtlass T. [6] proposed a hybrid model that each individual obeys the conventional swarm algorithm, but from time to time the DE is run which may move one individuals from a poorer area to a better area to continue the search.

In this paper, a novel hybrid global optimization method, termed PSODE, is introduced for application as a tool in solving challenging global optimization problems.

* Corresponding author.

PSODE is based on a two-population based scheme, in which the individuals of one population is enhanced by PSO and the individuals of the other population is evolved by DE. An information sharing mechanism is presented by the parallel simulation of PSO and DE. The interactions between the two populations influence the balance between exploration and exploitation and maintain some diversity in the whole population, even when it is approaching convergence, thus reducing the risk of convergence to local sub-optima.

The rest of the paper is organized as follows. Section 2 describes the PSO and DE. Section 3 motivates and describes the PSODE algorithm and gives the pseudocode for the algorithm. Section 4 defines the benchmark problems used for experimental comparison of the algorithms, and the experimental settings for each algorithm. Section 5 presents the results followed by conclusions in section 6.

2 Review of Standard PSO and DE

2.1 PSO

The fundament to the development of PSO is hypothesis that a potential solution to the optimization problem is treated as a bird without quality and volume, which we often call a particle, flying through the D -dimensional space, adjusting its position in search space according to its own experience and that of its neighbors.

The i th particle is represented as $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ in the D -dimensional space, where $x_{id} \in [l_d, u_d], d \in [1, D]$, l_d, u_d are the lower and upper bounds for the d th dimension, respectively. The rate of velocity for particle i is represented as $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$, is clamped to a maximum velocity v_{\max} , which is specified by the user. In each time step t , the particles are manipulated according to the following equations:

$$v_i(t+1) = w \times v_i(t) + R_1 c_1 (P_i - x_i(t)) + R_2 c_2 (P_g - x_i(t)) \quad (1)$$

$$x_i(t+1) = x_i(t) + v_i(t) \quad (2)$$

where R_1 and R_2 are random values between 0 and 1. c_1 and c_2 are acceleration constants, which control how far a particle will move in a single iteration. w is inertia weight, which often decreases linearly from about 0.9 to 0.4 during a run [7].

2.2 DE

DE technique combines simple arithmetic operators with the classical events of crossover, mutation and selection to evolve from a randomly generated starting population to a final solution.

DE-rand/1-exp scheme is recommended to be the first choice when trying to apply differential evolution to any given problem [4]. This particular version is adopted in our work, which is briefly described as follows. For a minimization problem,

i.e. $\min F(\bar{x})$, DE starts to work with a population of N candidate solutions, i.e. $\bar{x}_i^t, i = 1, 2, \dots, N$, where i indexes the population and t is the current generation.

For the mutation operation, a perturbed vector \bar{v}_i^t is generated according to

$$\bar{v}_i^t = \bar{x}_{r1}^t + F(\bar{x}_{r2}^t - \bar{x}_{r3}^t) \tag{3}$$

with random indexes $r1, r2, r3 \in \{1, 2, \dots, N\}$ and a scaling factor $F \in [0, 2]$.

For the crossover operation, the perturbed vector $\bar{v}_i^t = [v_{i1}, v_{i2}, \dots, v_{iD}]$ and target vector $\bar{x}_i^t = [x_{i1}, x_{i2}, \dots, x_{iD}]$ both are used to generate a trial vector $\bar{x}_i^{t'} = [x'_{i1}, x'_{i2}, \dots, x'_{iD}]$:

$$x'_{ij} = \begin{cases} v_{ij}, & \text{if } randb(j) \leq CR \text{ or } j = randr(i) \\ x_{ij}, & \text{if } randb(j) \geq CR \text{ and } j \neq randr(i) \end{cases} \tag{4}$$

where $j \in [1, D]$, $randb(j) \in [0, 1]$ is the j th evaluation of a uniform random number generator, $CR \in [0, 1]$ is the crossover constant. $randr(i) \in [1, 2, \dots, D]$ is a randomly chosen index which ensures that $\bar{x}_i^{t'}$ gets at least one parameter from \bar{v}_i^t .

For selection operation, a greedy scheme is performed:

$$\bar{x}_i^{t+1} = \begin{cases} \bar{x}_i^{t'}, & \text{if } \Phi(\bar{x}_i^{t'}) < \Phi(\bar{x}_i^t) \\ \bar{x}_i^t, & \text{otherwise} \end{cases} \tag{5}$$

where $\Phi(\bar{x})$ represents a fitness function.

3 PSODE Algorithm

In this paper, we propose a new algorithm based on PSO and DE. The original objective is to get benefits from both approaches. The major difference between Differential Evolution and Particle Swarm Optimization is how new individuals are generated. These new individuals generated on each generation are called offspring. It is caused by the selection schemes. Using DE only vectors will be admitted for the following generation that yields a smaller objective function value than the respective target vector. This is called a greedy selection scheme because no deteriorations with regards to the objective function value are possible. In contrast the PSO algorithm accepts all evolved particles, regardless of their objective function value.

In the basic PSO, all individuals are attracted by the best position found by themselves and the whole population. In this way the sharing of information among individuals is only achieved by employing the publicly available information P_g .

Table 1. Pseudocode for the PSODE algorithm

 Algorithm PSODE
Begin

Initialize all the populations

Divide them into two groups: P_1 and P_2

Evaluate the fitness value of each particle

Repeat**Do in parallel**Perform DE operation on P_1 Perform PSO operator on P_2 **End Do in parallel****Barrier synchronization** //wait for all processes to finishSelect the fittest local individual from P_1 Select the fittest local individual from P_2

Determine the global best in the whole population

Evaluate the fitness value of each particle

Until a terminate-condition is met**End**

Therefore, the population may lose diversity and is more likely to confine the search around local minima.

To solve the problem of diversity loss and premature convergence in the basic PSO model, we proposed a hybrid global optimization model. Our approach generates two population offspring individuals, one generated by the PSO mechanism and the other by DE one. There is mutual exchange of best particle information between two populations when they are executed in parallel. The idea behind the proposed algorithm is that the information can be transferred among individuals of different population that will help the individuals to avoid misjudging information and becoming trapped by poor local minima. Table 1 shows the pseudocode of the proposed hybrid algorithm PSODE.

4 Experiment Setting and Benchmark Problems

To investigate the performance of PSODE, we compared it to SPSO and DE in a set of benchmark optimization problems that are commonly used in literature [7, 8]. The benchmark problems used are a set of four non-linear functions, used as minimization problems, which present different difficulties to the optimization algorithms. They are Sphere function, Rosenbrock function, Rastrigrin function and Griewank function.

The parameters used for PSO are recommended from Shi and Eberhart [7]. The maximum velocity v_{\max} and minimum velocity v_{\min} for SPSO were set at half value

of the upper bound and lower bound, respectively. The acceleration constants c_1 and c_2 for PSO are both set as 2.0, the inertia weights are set to be $w_{\max} = 0.9$ and $w_{\min} = 0.4$. The DE parameters used here are $F = 0.8$ and $CR = 0.5$. For PSODE, the parameters, $c_1, c_2, w_{\max}, w_{\min}, F$ and CR , are all the same with those defined in PSO and DE. The population size is set as 40, and the dimension of the functions is equal to 10. A total of 20 runs for each experimental setting are conducted.

5 Experiment Results

SPSO, DE and PSODE were used to optimize the four benchmark functions using the settings presented in the previous paragraph. The results for the benchmark problem are shown in Table 2. Moreover, Figures 1-4 show the convergence graphs for the benchmark functions. All results below were reported as '0000e+000'.

Table 2. Results for all algorithms on benchmark problems

Function	Results	DE	PSO	PSODE
Sphere	Mean	8.3857e-038	1.8051e-045	2.2864e-051
	Std	3.5160e-074	1.6596e-089	2.3016e-101
Rosenbrock	Mean	2.8662 e+000	1.6916e+000	0.8804e+000
	Std	1.6892 e+000	1.7038e+003	1.2502e+000
Rastrigrin	Mean	2.0894e+000	3.3311e+000	7.9601e-001
	Std	1.4189e+000	2.9464e+000	5.8400e-001
Griewank	Mean	3.8496e-002	3.8822 e-002	3.1011e-002
	Std	7.8891e-004	6.0670 e-004	2.8418e-004

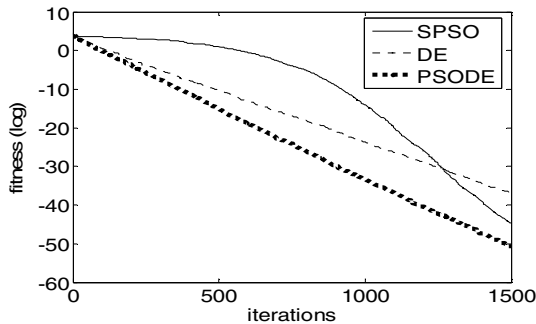


Fig. 1. Convergence graph for Sphere function

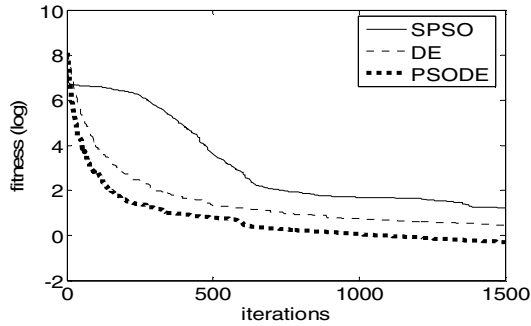


Fig. 2. Convergence graph for Rosenbrock function

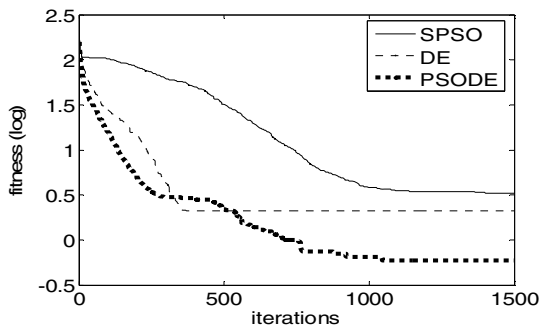


Fig. 3. Convergence graph for Rastrigrin function

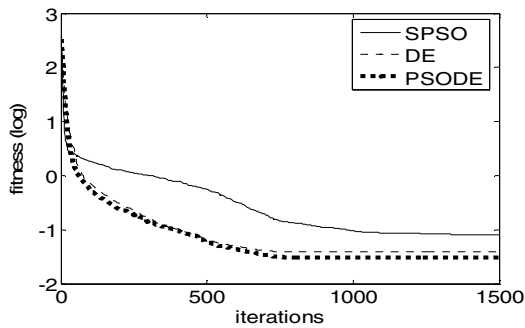


Fig. 4. Convergence graph for Griewank function

For the simplest function Sphere all the algorithms converge exponentially fast toward the fitness optimum. Since those problem is unimodal function, having only a single global minimum, fast convergence to the optimum is not a problem. However, only PSODE had particularly fast convergence, as can be seen from Figure 1. PSO converges slowly, but outperforms DE after 1300 iterations.

On function Rosenbrock, PSODE is superior to both PSO and DE. We may note that DE performs significantly better than PSO.

Function Rastrigrin is a highly multi-modal function when in 10 dimensions or less. Regarding this case DE converges very fast to good values near the optimum, while straggled with premature convergence after 400 generations. PSO performs worse than DE and PSODE. It stagnates and flats out with no further improvement after 1000 generations. On all of algorithms PSODE clearly performs best and gives consistently a near-optimum result.

On function Griewank PSO performs worse than DE and PSODE. PSODE performs nearly the same as DE at the first generations, but outperform DE after 800 generations.

It should be noted that with the PSODE method, the standard deviation of the final solution for 20 trails was found to be significantly low on all of the functions compared with DE and PSO, as show in Table 2. This illustrated that the results generated by PSODE is robust to all of the benchmark problems.

6 Conclusions and Future Work

In this paper a new hybrid optimization algorithm is presented which is based on the integration of PSO and DE. The proposed algorithm is simple in concept and no extra extra parameters are introduced. Four benchmark functions has been used to test PSODE in comparison with PSO and DE. Among them, two functions were unimodal and two were multimodal.

For the multimodal functions PSODE found better results than those generated by the other two methods PSO and DE. For the unimodal functions, of which the convergence rate is more important than the final results, our PSODE outperformed the other two algorithms in terms of accuracy and convergence rate. It can be concluded that by combing the two methods, the advantages of both methods are exploited to produce a hybrid optimization method which is both robust and fast.

Because PSO and DE are executed in parallel in our proposed algorithm, future work is focused on simulating PSODE algorithm by a parallel computer. In addition, different hybrid models of PSO and DE algorithm will be studied.

Acknowledgements. This work is supported by the National Natural Science Foundation of China under Grant No. 70271014.

References

1. Eberhart, R.C., Kennedy, J.: A New Optimizer Using Particle Swarm Theory. In: 6th IEEE international symposium on Micromachine and Human Science, pp. 39–43. IEEE Press, Piscataway (1995)
2. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: Proceeding of IEEE International Conference on Neural Networks, pp. 194–1948. IEEE Press, Piscataway (1995)
3. Kennedy, J., Eberhart, R.C., Shi, Y.: Swarm Intelligence. Morgan Kaufmann Publishers, San Francisco (2001)

4. Kenneth, V.: Price, An Introduction to Differential Evolution. In: Corne, D., Dorigo, M., Glover, F. (eds.) *New Ideas in Optimization*, pp. 79–108. McGraw-Hill, London (1999)
5. Zhang, W.J., Xie, X.F.: DEPSO: Hybrid Particle Swarm with Differential Evolution Operator. In: *Proc. of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 3816–3821. IEEE Press, Washington (2003)
6. Hendtlass, T.: A Combined Swarm Differential Evolution Algorithm for Optimization Problems. In: Monostori, L., Váncza, J., Ali, M. (eds.) *IEA/AIE 2001. LNCS (LNAI)*, vol. 2070, pp. 11–18. Springer, Heidelberg (2001)
7. Shi, Y., Eberhart, R.C.: A Modified Particle Swarm Optimizer. In: *Proceeding of the 1998 IEEE International Conference on Evolutionary Computation*, pp. 69–73. IEEE Press, Piscataway (1998)
8. Niu, B., Zhu, Y.L., He, X.X., Wu, H.: MCPSO: A Multi-Swarm Cooperative Particle Swarm Optimizer. *Applied Mathematics and Computation* 185, 1050–1062 (2007)

A Parallel Multi-algorithm Solver for Dynamic Multi-Objective TSP (DMO-TSP)

Ming Yang¹, Zhou Kang^{2,*}, and Lishan Kang^{1,2}

¹ School of Computer Science, China University of Geosciences, Wuhan, 430074, China

² Computation Center, Wuhan University, Wuhan, 430072, China

yangming0702@gmail.com, kang_zh@sohu.com, kang_whu@yahoo.com

Abstract. Dynamic Multi-Objective TSP (DMO-TSP) proposed as a theoretical model of mobile communication network in 2004 is an NP-hard problem. The problem dynamically changes the characteristics of its objectives, the conflict degrees between its objectives and the number of its cities. In fact, a Dynamic Multi-Objective TSP is not a single optimization problem, but a diverse set of optimization problems. The No Free Lunch Theorems in optimization and numerical experiments have demonstrated that it is impossible to develop a single evolutionary algorithm for population evolution that is always efficient and effective for solving such an extremely complicated diverse set of optimization problems. In this paper, a parallelized form of the multi-algorithm co-evolution strategy (MACS) for DMO-TSP called synchronized parallel multi-algorithm solver is proposed, because the MACS solver can just continuously track the moving Pareto front of small size (about 100 cities) DMO-TSP with two objectives in lower degree of conflict. It is hoped that the synchronized parallel multi-algorithm solver can be used to track the moving Pareto front efficiently for larger size DMO-TSP with higher conflict degrees between objectives by distributed parallel computer systems with shared memory.

Keywords: dynamic multi-objective TSP; parallel multi-algorithm; Pareto optimal front.

1 Introduction

In many application areas of sciences, such as computational chemistry, biology, bioinformatics, economics, computational science, geophysics, and environmental science, there are many optimization problems solved by evolutionary optimization algorithms. Many of these problems are time depend and have many objectives. For example, in mobile communication network design, one has to consider some goals simultaneously, such as the shortest communication route, the minimum communication time, the lowest risk, the minimum cost, and many other objectives in a dynamic environment that lead to solve the dynamic multi-objective TSP (DMO-TSP). How to track a moving set of Pareto optimal solutions of a dynamic multi-objective TSP (DMO-TSP) is a very complicated and hard task.

* Corresponding author.

Dynamic multi-objective TSP (DMO-TSP) is a combination of dynamic TSP (D-TSP) and multi-objective TSP (MO-TSP). Dynamic multi-objective TSP (DMO-TSP) was first proposed by the authors as a project of National Nature Science Foundation of China in 2004. It was used as a 3-D mathematical model of space communication network optimization model which consists of ground-grid, air-grid and space-grid. At that time, there were no any reports of theoretical and numerical results on DMO-TSP. Because DMO-TSP demands to track the dynamic Pareto optimal front in real time, so it is a grant challenge to computer scientists and mathematicians. Steadies of DMO-TSP are of great theoretical and practical significances.

A wide variety of algorithms have been proposed for TSP and D-TSP, such as LK algorithm[1], LKH algorithm [2], Inver-Over operator [3,4] and elastic net method [11] for TSP, and ant algorithms [6-8], competitive algorithms (on-line algorithms) [9], dynamic Inver-Over evolutionary algorithms [10] and some effective algorithms composed of several operators [10] for D-TSP. There are also optimization algorithms [12-14], which are based on Pareto optimality [15], for multi-objective TSP. But these algorithms can be just used to solve some small size MO-TSP. This is also true to dynamic Inver-Over algorithm for D-TSP. These algorithms can not meet real-time demands. For multi-objective problems, the number of Pareto optimal solutions depends on the conflict degree between objectives. For dynamic multi-objective TSP, the conflict degree between objectives and the number of cities change with time. In fact, a dynamic multi-objective TSP is not a single optimization problem, but a diverse set of optimization problems. The No Free Lunch Theorems in Optimization [16] and numerical experiments have demonstrated that it is impossible to develop a single evolutionary algorithm for population evolution that is always efficient and effective for solving such an extremely complicated diverse set of optimization problems.

There have been some studies indicated that algorithms combining several methods [18][19] especially, genetically adaptive multi-method [24] can be more effective for solving multi-objective function optimization problems, but there is no such a kind of algorithms for solving DMO-TSP. Recently, Yang et al. [26] proposed an Multi-Algorithm Solver called multi-algorithm co-evolution strategy (MACS) for DMO-TSP, which can be used for solving small size DMO-TSP with low degree of conflict objectives in real time. But it is difficult to track moving Pareto optimal front of larger size DMO-TSP in real time. In this paper, a synchronized parallel multi-algorithm solver is proposed for solving these kinds of problems by multi-processor systems with shared memory.

2 What Is a Dynamic Multi-Objective TSP

The cost matrices between cities in DMO-TSP change with time. A DMO-TSP can be described by the following equations:

$$D_k(t) = \{d_{ij}^k(t)\}_{n(t) \times n(t)}, \quad k = 1, 2, \dots, m. \quad (1)$$

where $d_{ij}^k(t)$ is the cost of objective k from $city_i$ to $city_j$ at time t , $n(t)$ is the number of cities at time t , m is the number of objectives and t is the real word time. The length of the route $\pi(t)$ at time t for objective k is

$$f_k(\pi(t)) = \sum_{j=1}^{n(t)} d_{\pi_j \pi_{j+1}}^k(t), \quad k = 1, 2, \dots, m, \quad \pi_{n(t)+1} = \pi_1. \quad (2)$$

Given m objectives, $f_k(\pi(t))$, $k = 1, 2, \dots, m$, for route $\pi^*(t)$, if not exist any route $\pi(t)$ to make at least one of the following inequalities strictly true, then $\pi^*(t)$ is a dynamic Pareto optimal solution at time t .

$$f_k(\pi(t)) \leq f_k(\pi^*(t)), \quad k = 1, 2, \dots, m. \quad (3)$$

where at least one inequality strictly sets up.

$S^*(t)$, a set composed of all Pareto optimal solutions at time t , is the Pareto optimal set at time t . $F^*(t)$, a set composed of all images of Pareto optimal solutions in the objective space, is the Pareto optimal front. $S^*(t)$ and $F^*(t)$ are finite sets. The larger conflict degree of objectives, the larger number of Pareto optimal solutions is.

For detecting the changes of cost matrices with time t , we choose a series of sampling points of time: $t_0, t_1, t_2, \dots, t_N$, where $t_{i+1} = t_i + \Delta t$. The changed information is stored in a Dlist (Data List). Thus, a DMOTSP becomes a series of static multi-objective TSPs with different characteristics.

The task for solving DMO-TSP becomes to solve a series of different MO-TSPs in time intervals $[t_0, t_1], [t_1, t_2], \dots, [t_{N-1}, t_N]$.

3 What Algorithms Should Be Selected

The multi-algorithm solver consists of seven algorithms: Gene-Inver-Over, Pareto-Inver-Over, Objective Best Individual Crossover (OBIC), Objective-PMX, Pareto-PMX, Dynamic Elastic Relaxation and Domain Decomposition Algorithms. When the problem's environment changes, Dynamic Elastic Relaxation algorithm is used to optimize the population locally, then Gene-Inver-Over and Pareto-Inver-over algorithms are used to optimize the population globally. The OBIC is used at the later stage of evolution for optimizing the best individual for each objective, while Objective-PMX is used to optimize the rest individuals with the best individual of each objective and make the population close to the Pareto optimal front. After these algorithms work together co-operatively, MACS generates offspring which make the Pareto set diversity and their images distribute on the Pareto optimal front evenly. These algorithms are described as follows:

A gene pool is constructed as in [20].

Suppose that R_i is an individual of population and R' denotes its offspring individual. The Gene-Inver-Over algorithm is described as follows.

(1) **Gene-Inver-Over Algorithm:**

for(each objective k)

{

$R' := R_i$;

select a city c randomly from R' ;

repeat

```

{
  select a city  $c'$  randomly from the gene pool of  $c$  for objective  $k$ ;
  if (the 'next' city to  $c$  in  $R'$  is  $c'$ ) then exit from loop;
  else inverse the fragment  $r'=(c,\dots,c')$  of  $R'$ ;
   $c := c'$ ;
}
if ( $R' \succeq R_i$ ) then  $R_i := R'$ ;
if ( $R' \parallel R_i$ ) then  $R'$  is selected into population;
}

```

The Pareto-Inver-Over algorithm is described as follows.

(2) **Pareto-Inver-Over Algorithm:**

```

 $R' := R_i$ ;
select a city  $c$  randomly from  $R'$ ;
repeat
{
  select an individual  $R^p$  randomly from Pareto set;
  The 'next' city to  $c$  in  $R^p$  is assigned to  $c'$ ;
  if (the 'next' city to  $c$  in  $R'$  is  $c'$ ) then exit from loop;
  else inverse the fragment cities  $r'=(c,\dots,c')$  of  $R'$ ;
   $c^i := c'$ ;
}
if ( $R' \succeq R_i$ ) then  $R_i := R'$ ;
if ( $R' \parallel R_i$ ) then  $R'$  is added into population;

```

Suppose that R_k^* is the best individual for objective k and R_i is an individual in population.

OBIC can be used at the later stage of evolution for optimizing the best individual of each objective. The OBIC algorithm is described as follows:

(3) **OBIC Algorithm:**

```

for(each objective  $k$ )
{
  select a gene fragment  $r^* = \{ c_i, \dots, c_j \}$  randomly from  $R_k^*$ ;
  for(individual  $R_i$  of population)
  {
    if (exist a better gene fragment  $r' = \{ c_i, \dots, c_j \}$  in  $R_i$  then replace  $r^*$  of  $R_k^*$  with  $r'$ ;
  }
}

```

The Objective-PMX Algorithm is described as follows:

(4) **Objective-PMX Algorithm:**

```

for(each objective  $k$ )
{
  for (individual  $R_i$  of population)

```



```

    {
        select  $a$  gene fragment randomly from  $R_k^*$  and mapped it into  $R_i$  by partially mapped (PMX) method [23];
        if( $R' \succeq R_i$ ) then  $R_i := R'$ ;
        if( $R' \parallel R$ ) then  $R'$  is added into population;
    }
}

```

The Pareto-PMX is described as follows:

(5) Pareto-PMX Algorithm:

```

for (each individual  $R^*$  of Pareto set)
{
    select a gene fragment randomly from  $R^*$  and mapped it into  $R_i$  by PMX method;
    if ( $R' \succeq R_i$ ) then  $R_i := R'$ ;
    if ( $R' \parallel R_i$ ) then  $R'$  is added into population;
}

```

Dynamic elastic relaxation algorithm is composed of Delete Operator, Insert Operator and Change Operator. When the number of cities and the positions of cities are changing with time t , the dynamic elastic relaxation algorithm can be used to get an approximate optimal route quickly.

(6) Dynamic Elastic Relaxation Algorithm:

```

Delete the node C, link the cities adjacent to C;
Find the nearest node  $C^*$  to C in route;
Insert C into the route besides  $C^*$  that minimize the increasing length;

```

(7) Domain Decomposition method

The network space $S \subset R^3$ in which all cities (nodes) may emerge dynamically is decomposed into j overlapping cubes where each cube is defined by a centroid and side pair: $S_j=(c_j, r_j)$, such that each cube contains about 50 cities. Usually the cities are uniformly distributed in S and r_j can be taken as a constant. Assume that a multi-processor system with shared memory is used to implement the following parallel Domain Decomposition Algorithm:

```

for(each sub-domain  $s_j$ ) do in parallel with  $j$  processors
    Dynamic Elastic Relaxation Algorithm;
    construct gene pool;
end do

```

Suppose at time t that the population size is n , the number of approximate Pareto solutions is ν and the set of the individuals which have the best fitness for every objective in current population is W . Denote the individual in population by R , its offspring by R' , “ R' dominates R ” by $R' \succeq R$, and “ R' and R are not comparable” by $R' \parallel R$. In the evolution process, if $R' \succeq R$, then $R := R'$. If $R' \parallel R$, then R and R' are selected into offspring population. The value of ν may larger than n . The Selection Rules for Generating Offspring Population are described as follows:

(8) **Rule A**

- a) If $v = n$, then take the individuals of Pareto set as the offspring population;
- b) If $v > n$, then search the two individuals in Pareto set the distance between which in the objective space is the minimum and they do not belong to W , suppose that they are R_1 and R_2 . For R_1 (R_2), search the minimum distance D_1 (D_2). If $D_1 < D_2$, then delete R_1 from Pareto set; else delete R_2 . Repeat the above process until $v = n$. Finally take the individuals of Pareto set as the new population;
- c) If $v < n$, suppose that $n-v = u$, then select randomly u individuals which do not belong to Pareto set from population and take these u individuals and the ones of Pareto set together as the offspring population.

4 Synchronized Parallel Algorithm for DMO-TSP

Suppose that Dlist is a list for storing the information of cities' state change which is updated at every sampling time point, h and h_j are the serial number of iterations. The synchronized parallel algorithm can be described as follows.

Synchronized Parallel Algorithm:

- Step 1. initialize population $P = \{R_1, R_2, \dots, R_n\}$ randomly; $t:=0$;
- Step 2. $h:=0$; **if** Dlist is empty **then goto** step 4;
- Step 3. **for** each sub-domain s_i **do** in parallel with j processors
 Dynamic Elastic Algorithm; reconstruct gene pool;
 end do
- Step 4. **if** $h>H$ **then goto** step 10;
- Step 5. **for** each individual R_i of population **do** in parallel with n processors
 Gene-Inver-Over;
 Pareto-Inver-Over;
 end do
- Step 6. generate offspring population according rule A;
 if $h \leq p_1 * H$ **then goto** step 8;
- Step 7. **for** each individual R_i of population **do** in parallel with n processors
 OBIC;
 end do
- Step 8. **for** each individual R_i of population **do** in parallel with n processors
 Objective-PMX;
 end do
- Step 9. generate offspring population according rule A;
 $h:=h+1$; **goto** step 4;
- Step 10. $h_j:=0$;
- Step 11. **if** $h_j>H_j$ **then goto** step 14;
- Step 12. **for** each individual R_i of population **do** in parallel with n processors
 Pareto-PMX;
 end do
- Step 13. generate offspring population according rule A;
 $h_j:=h_j+1$; **goto** step 11;
- Step 14. sampling ; $t := t + 1$; **if** $t \leq T$ **then goto** step 2;
- Step 15. **end**

where population: p , Pareto set: PS , gene pool: GP and $Dlist$ are global variables which are stored in shared memory. A host processor is used as the controller at synchronized points.

The synchronized parallel program chart is described as follows:

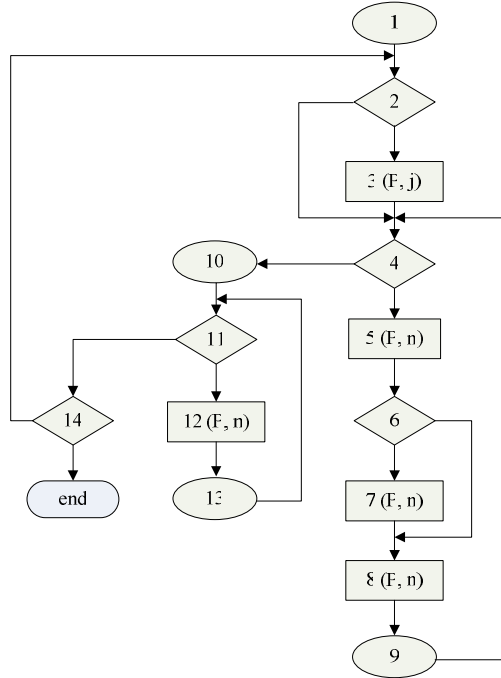


Fig. 1. The Synchronized Parallel Program

where i and i denote that step i is performed by host processor, while $i(P, n)$ denotes that step i is performed asynchronously with n processors.

5 Numerical Simulation

The experimental DMO-TSP model is a 3-D mobile communication network which consists of 149 nodes, including China 145 cities and 4 satellites, one of them is a geo-stationary satellite over China and other three mobile satellites, two in polar orbit and one in equatorial orbit. There are two objectives, a distance matrix and a cost matrix, which depend on time. For more detail, see [26].

Parameters are as follows: $n = 30$, $j = 6$, $H = 200$, $p_1=0.6$, $H_1= 50$, $T =120$.

Experiments are simulated by an IBM ThinkPad with processor 1.50GHz, RAM 512MB, which is simulated by software and is not the real parallel solver.

The simulated results are shown by figures.

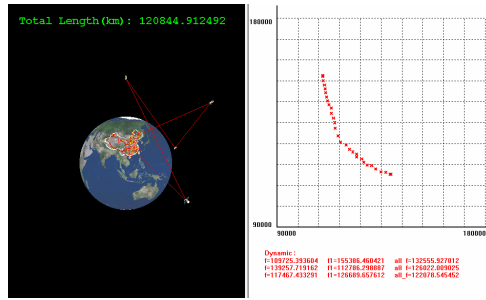


Fig. 2. The Dynamic Pareto Optimal Front at $t=30$

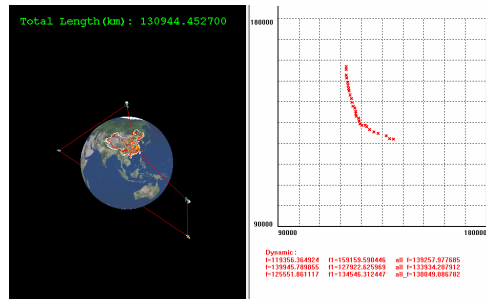


Fig. 3. The Dynamic Pareto Optimal Front at $t=60$

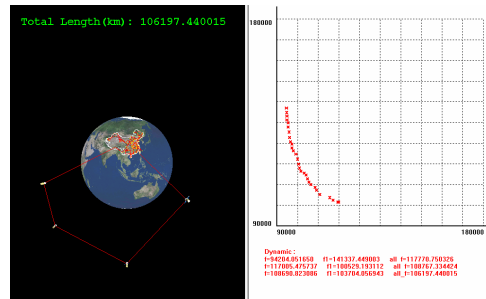


Fig. 4. The Dynamic Pareto Optimal Front at $t=90$

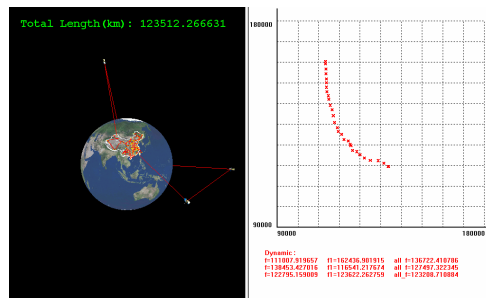


Fig. 5. The Dynamic Pareto Optimal Front at $t=120$

6 Conclusion

A multi-processor system with shared memory can be used to perform an asynchronous parallel algorithm efficiently. In our synchronized parallel multi-algorithm solver, there are five parallel sub-algorithms, four of which can be asynchronously performed with 30 processors, another one can be asynchronously performed with 6 processors. We use one processor worked as a host for controlling synchronization. It must have high speed-up ratio. But our experiments are just simulated by an IBM ThinkPad. So this conclusion remains to be confirmed by using a distributed parallel computer system to do more numerical experiments.

References

1. Lin, S., Kernighan, B.W.: An Effective Heuristic Algorithm for the Traveling Salesman Problem. *Operations Research* 21, 498–516 (1973)
2. Helsgaun, K.: An Effective Implementation of the Lin-Kernighan Traveling Salesman Heuristic. *Eur. J. Oper. Res.* 126, 106–130 (2000)
3. Guo, T.: *Evolutionary Computation and Optimization* (Ph.D Thesis). Wuhan University, Wuhan (May 2000)
4. Guo, T., Michalewicz, Z.: Inver-Over Operator for the TSP. In: Eiben, A.E., Bäck, T., Schoenauer, M., Schwefel, H.-P. (eds.) *PPSN 1998*. LNCS, vol. 1498, pp. 803–812. Springer, Heidelberg (1998)
5. Psaraftis, H.N.: Dynamic Vehicle Routing Problems. In: Golden, B.L., Assad, A.A. (eds.) *Vehicle Routing: Methods and Studies*, pp. 223–248. Elsevier Science Publishers, Amsterdam (1988)
6. Guntsch, M., Branke, J., Middendorf, M., Schmeck, H.: ACO Strategies for Dynamic TSP. In: Dorriago, M., et al. (eds.) *Abstract Proceedings of ANTS 2000*, pp. 59–62 (2000)
7. Guntsch, M., Middendorf, M.: Pheromone Modification Strategies for Ant Algorithms Applied to Dynamic TSP. In: Boers, E.J.W., Gottlieb, J., Lanzi, P.L., Smith, R.E., Cagnoni, S., Hart, E., Raidl, G.R., Tijink, H. (eds.) *EvoIASP 2001, EvoWorkshops 2001, EvoFlight 2001, EvoSTIM 2001, EvoCOP 2001, and EvoLearn 2001*. LNCS, vol. 2037, pp. 213–220. Springer, Heidelberg (2001)
8. Guntsch, M., Middendorf, M., Schmeck, H.: An Ant Colony Optimization Approach to Dynamic TSP. In: *Proceedings of the GECCO 2001*, pp. 860–867. Morgan Kaufmann, San Francisco (2001)
9. Ausiello, G., Feunteun, E., Leonardi, S., Stougie, L., Talamo, M.: Algorithms for the On-line Traveling Salesman. *Algorithmica* 29(4), 560–581 (2001)
10. Kang, L.S., Zhou, A., McKay, B., Li, Y., Kang, Z.: Benchmarking Algorithms for Dynamic Travelling Salesman Problems. In: *Proceedings of the Congress on Evolutionary Computation*, Portland, Oregon (2004)
11. Durbin, R., Willshaw, D.: An Analogue Approach to the Traveling Salesman Problem Using an Elastic Net Method. *Nature* 326(6114), 689–691 (1987)
12. Yan, Z.Y., Zhang, L.H., Kang, L.S., Lin, G.M.: A New MOEA for Multi-objective TSP and Its Convergence Property Analysis. In: Fonseca, C.M., Fleming, P.J., Zitzler, E., Deb, K., Thiele, L. (eds.) *EMO 2003*. LNCS, vol. 2632, pp. 342–354. Springer, Heidelberg (2003)

13. Ji, Z.W., Chen, A., Kitti, S.: Finding Multi-Objective Paths in Stochastic Networks: A Simulation-based Genetic Algorithm Approach. In: Congress on Evolution Computation (CEC 2004), pp. 174–180. IEEE Press, Los Alamitos (2004)
14. Marwaha, S., Srinivasan, D., Tham, C.K., et al.: Evolutionary Fuzzy Multi-Objective Routing For Wireless Mobile Ad Hoc Networks. In: Congress on Evolution Computation (CEC 2004), pp. 1964–1971. IEEE Press, Los Alamitos (2004)
15. Pareto, V.: Cours D' Economie Politique, vol. I. F. Rouge, Lausanne (1896)
16. Wolpert, D.H., MacReady, W.G.: No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* 1(1), 67–82 (1997)
17. Kang, L.S., Jin, H.Y., et al.: The Split Method for Solving High-dimensional Partial Differential Equations, pp. 160–196. Science and Technology of Shanghai Press (1990)
18. Peter, M., Freisleben, B.: Genetic Local Search for the TSP: New Results. In: Proceedings of the 1997 IEEE International Conference on Evolutionary Computation, pp. 164–259. IEEE Press, NJ (1997)
19. Baraglia, R., Hidalgo, J.I., Perego, R.: A Hybrid Heuristic for the Traveling Salesman Problem. *IEEE Transactions on Evolutionary Computation* 5(6), 613–622 (2001)
20. Yang, Hui, Kang, L.S., Chen, Y.P.: A Gene-pool Based Genetic Algorithm for TSP. *Wuhan University Journal of Nature Sciences* 8(1B), 217–223 (2003)
21. Volgenant, T., Jonker, R.: The Symmetric Traveling Salesman Problem and Edge Exchange i Minimal 1-Trees. *Eur. J. Oper. Res.* 12, 394–403 (1983)
22. TSP Library (2006), <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/tsp>
23. Goldberg, D.E., Lingle, R.: Alletes, Loci, and the TSP. In: Proceedings of the First International Conference on Genetic Algorithms. Lawrence Erlbaum Associates, Hilldale, pp. 154–159 (1985)
24. Vrugt, J.A., Robinsion, B.A.: Improved evolutionary optimization from genetically adaptive multi-method search. *Proceedings of the National Academy of Sciences of the United States of America* 104, 708–711 (2007)
25. Corne, D., Knowles, J.: Techniques for Highly Multi-objective Optimization: Some Non-dominated Points are Better than Others. In: Proceedings of GECCO, pp. 773–780 (2007)
26. Yang, M., Kang, L.S., Guan, J.: Multi-Algorithm Co-evolution Strategy for Dynamic Multi-Objective TSP. In: Proceedings of the Congress on Evolutionary Computation (CEC 2008), Hong Kong (2008)

A Virtual Three-Dimension Cellular Automata Pseudorandom Number Generator Based on the Moore Neighborhood Method

Sang-Ho Shin¹, Geum-Dal Park², and Kee-Young Yoo^{1,*}

¹ Department of Computer Engineering, Kyungpook National University, 1370, Sankyuk-Dong, Buk-Gu, Daegu, 702-701, South Korea

² School of Electrical Engineering and Computer Science, Kyungpook National University, 1370, Sankyuk-Dong, Buk-Gu, Daegu, 702-701, South Korea
{shshin80,machunru2}@infosec.knu.ac.kr, yook@knu.ac.kr

Abstract. The security of a One-time Pad Cryptography system depends on the keystream generator, which has been studied to produce a high randomness quality over the last thirty years. A Cellular Automata (CA) Pseudorandom Number Generator (PRNG) is more efficiently implemented rather than LFSR, Linear Congruential generator, Fibonacci generator, etc.. Moreover, a CA structure-based PRNG is highly regular and simpler than previous PRNGs. Accordingly, we propose a new PRNG based on a virtual three-dimension (3-D) CA with the Moore neighborhood structure. In order to evaluate the quality of randomness, the ENT and the DIEHARD test suites are used. The results of these tests show that the quality of randomness is better than previous PRNGs.

Keywords: Symmetric Key Cryptography, One-time Pad, Cellular Automata (CA), Pseudorandom Number Generator (PRNG).

1 Introduction

Recently, with the expansion of the network and Internet, our lives are more comfortable than previously. These include many tasks, such as sending e-mails, searching information, shopping, etc.. Internet security becomes more and more important. In order to transfer secret data (or information) over Internet securely. A Cryptography technique is a type of secret method which is a technique protecting the data from any wrong access by transforming important data into a senseless bitstream (such as a trust random bitstream).

The keystream generator based on CA is a new prospective method designed to generate pseudo-random numbers. It has been extensively studied over the past decades for its convenient implementation by means of self-reproduction and self-repair [2,3,4]. Recent interest has been focused on the two-dimensional (2-D) CA PRNGs [1,5,6], since, statistically, the point has been established that the quality of randomness of a 2-D CA is significantly better than a 1-D CA [8].

* Corresponding author. (Tel.: +82-53-950-5553; Fax.: +82-53-957-4846).

However, taking into account the design complexity and computation efficiency, it is quite difficult to conclude which one is better. In this paper, a new PRNG-virtual 3-D cellular automata PRNG is proposed with the Moore neighborhood (V3-DCA PRNG) which obtains a higher the quality of randomness. Based on the observation of the evolved V3-DCA PRNGs, we find that the randomness of V3-DCA structures for PRNGs are affected by their structures. In the meantime, the evolved V3-DCA PRNG is compared with previous PRNGs [1,5,6] to check the quality of randomness. The evolved V3-DCA PRNG generates a high quality of randomness which is passed by DIEHARD test suite [9].

This paper is organized as follows. In Section 2, related work is briefly reviewed. The V-3DCA PRNG is proposed, in Section 3. Section 4, shows the experimental results. Section 5 provides a conclusion.

2 Related Works

2.1 Two-Dimension Cellular Automata

CA is a dynamic system in which space and time are discrete. A CA consists of an array of cells, each of which can be in one of a finite number of possible states, updated synchronously in discrete time steps, according to a local, identical interaction rule. A 2-D CA is a generalization of a 1-D CA, where the cells are arranged in a two-dimensional grid with connections in combination the neighboring cells. For a 2-D CA, the following types of cellular neighborhoods are usually considered; five cells, consisting of one cell in combination with its four immediate non-diagonal neighbors (also known as the von Neumann neighborhood); and nine cells, consisting of one cell along with its eight surrounding neighbors (also known as the Moore neighborhood). In this paper, a type of Moore neighborhood is proposed, which considers nine-neighborhoods that each consist of self, top, bottom, left, right, top-left, top-right, bottom-left and bottom-right.

The next state $s(t)$ of (i, j) -th cell of a 2-D CA is given by

$$s_{i,j}(t+1) = f\left(\bigcup_{k,l=-1}^1 s_{i+k,j+l}(t)\right). \quad (1)$$

Since f is a Boolean function composed of nine variables, there are $2^9 = 512$ distinct neighborhood configurations. In order to express a transition rule of a 2-D CA in a manner similar to a 1-D CA, 512-bits are considered, which are almost impossible to practically manage.

Next, choosing the Boundary is considered. Generally, a Periodic Boundary CA (PBCA) is frequently applied, resulting in a circular grid for a 1-D case and in a doughnut for the 2-D case. A Null Boundary CA (NBCA) can also be utilized, in which the grid is surrounded by an outer layer of cells of zero. In this paper, a PBCA is selected in order to raise the quality of randomness.

2.2 Previous Works

In this section, previous PRNGs are described. Tomassini et al. proposed a 2-D CA PRNG based on 8×8 structure [1] and Guan et al. proposed 2-D CA PRNGs based on Asymmetric neighborhood and lattice structure [5,6]. In its final phases, these PRNGs are constructed by using the evolutionary method is called genetic algorithms [12]. A genetic algorithm has operators that consist of crossover, mutation, and replacement-swapping [7] and has a fitness (entropy) function. Iterating this procedure, the genetic algorithm may eventually find an acceptable solution. Hence, a genetic algorithm provides a high randomness quality, but has problems including high area and time complexity.

In Tomassini et al.'s PRNG [1], every cell of the evolving CA assigns a fitness function value and rule number which determines the result of the evolved cell by the fitness function value. Finally, a total of 30 experiments were performed and seven rules of good randomness, which are 31, 47, 55, 59, 61, 62, and 63, respectively, were found. Also, to raise the quality of randomness, a genetic algorithm was used. A total of 20 DIEHARD test suite [9] for some rules (55, 59, 61, and 63) were performed. However, it was determined that the result of these test were poor [13].

Guan et al.'s PRNGs [5,6] are constructed from an asymmetric neighborhood (5×10) and 2-D lattice (15×3 , 7×6) CA structure for reducing the number of cells. However, there is a structural problem. The PRNG structures used the same rules with the exception of the first and last columns. The hybrid CA structure can provide better performance [3]. Consequently, it is known that the pass rate of DIEHARD test for Guan et al.'s PRNGs is less than Tomassini et al.'s PRNG.

3 Proposed V3-DCA PRNG

The V3-DCA PRNG produces a high quality random number sequence having a low hardware construction cost. In order to achieve a higher the quality of randomness, we choose a distinguishable method from previous schemes [1,5,6]. In this chapter, the proposed PRNG is explained.

3.1 Rule Numbering Function (*RNF*)

In order to describe the proposed V3-DCA PRNG, the following rule numbering function (*RNF*) is introduced. Generally, Moore neighborhood method utilizes a 2-D CA with nine-neighborhood. The reference cell number value of this neighborhood is greater than that of von Neumann neighborhood. In regards to hardware implementation cost, Moore neighborhood method is be expensive more than von Neumann neighborhood method. In this paper, to raise more randomness quality, Moore neighborhood method was used. Concept of proposed V3-DCA PRNG is the Moore neighborhood-based method and equation (1) that there are 2^9 distinct neighborhood configuration. Hence, to express a rule of 2-D CA in a manner similar to 1-D CA, 512-bits are required (which is almost

impossible to manage in practice). Due to their interesting properties and ease of characterization, a CA with a linear neighborhood relationship (XOR) was only considered. The rule can be expressed as a 9-bit in proposed PRNG and Figure 1 shows the method of *RNF*. In Figure 1, *RNF* value (*RNFv*) consists of 64-bit sequence (that is, *RNFv* is rule in the proposed PRNG). *RNFv* of *i*th cell is decimal that this value converts binary values into decimal.

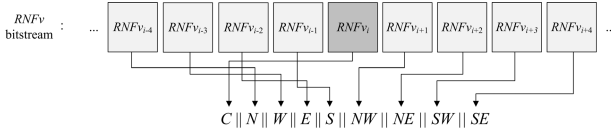


Fig. 1. The rule numbering method

Let $s_{i,j}(t)$ be the state of the cell at row i and column j in time t . Its state at the next time step, $s_{i,j}(t + 1)$ is then computed as a following equation (2):

$$s_{i,j}(t + 1) = C \cdot s_{i,j}(t) \oplus DC \oplus NDC, \tag{2}$$

where,

$$DC = N \cdot s_{i-1,j}(t) \oplus S \cdot s_{i+1,j}(t) \oplus E \cdot s_{i,j+1}(t) \oplus W \cdot s_{i,j-1}(t),$$

$$NDC = NW \cdot s_{i-1,j-1}(t) \oplus NE \cdot s_{i-1,j+1}(t) \oplus SW \cdot s_{i+1,j-1}(t) \oplus SE \cdot s_{i+1,j+1}(t),$$

\oplus and \cdot are the Boolean operations XOR and AND, respectively. $C, N, S, W, E, NW, NE, SW$ and SE are binary variables and denote whether the respective neighboring cell state is taken into account (a value of) 1 or not (a value of) 0.

3.2 Cell Position Function (CPF)

In this section, the cell position function (*CPF*) is explained. To decrease the correlation coefficient of generated global states, the *CPF* was utilized. Generally, the Moore neighborhood in 2-D CA refers the nine-bit at t -time step. In this paper, the arbitrary three-bit in three-global state ($G^{(t-2)}, G^{(t-1)}$ and $G^{(t)}$) were referred and cases of *CPF* method are as following.

There are 8 methods and the position value of the total number of method by 3-bit (3-bit is *CPF* value (*CPFv*)) was represented. Figure 2 shows cell position method. In Figure XX, if $CPFv_{i-1}$ is 0, then a part of each state row was selected. Otherwise ($CPFv_{i-1}$ is 1), then a part of each state column was selected (except the case of three-bit is 000 or 111). If *CPFv* is 000 or 111, then non-diagonal cells were selected. Figure 3 shows all cases of cell position function.

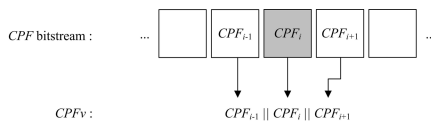


Fig. 2. The cell position method

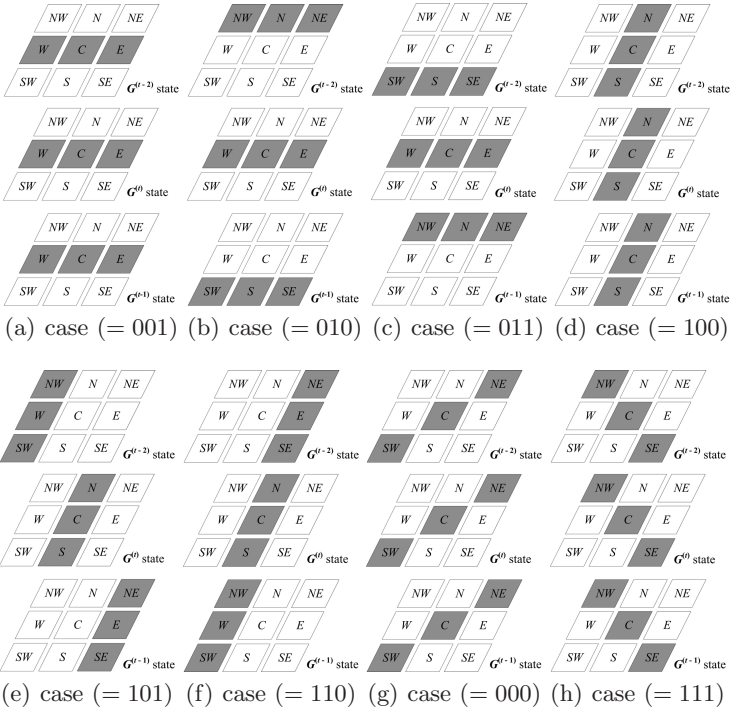


Fig. 3. All cases of cell position function

3.3 Process of Virtual Three-Dimension CA PRNG

One round consists of three phases in proposed scheme. The step-by-step process is summarized as follows.

Initial phase. The 192-bit initial seed is randomly generated. The seed then divides into three parts, from 0^{th} to 63^{th} ($= seed1$), from 64^{th} to 127^{th} ($= seed2$) and from 128^{th} to $final\ bit$ ($= seed3$). The initial values $seed1$, $seed2$, and $seed3$ are stored in $G^{(-2)}$ (Global states of an 8×8 2-D CA), $G^{(-1)}$ and $G^{(0)}$, respectively. The $G^{(-2)} \oplus G^{(-1)}$ is calculated and stored in the initial *RNF* value (*RNFv*) and the $G^{(-1)} \oplus G^{(-0)}$ is calculated and stored in the initial *CPF* value (*CPFv*).

Evolving phase in two-dimension CA. The rule $r_{i,j}$ is decided using *RNF* and *CPF* which decide with self, diagonal 4-bits and non-diagonal 4-bits and 3-bits *CPFv*. A next state $G^{(t+1)}$ by $r_{i,j}$, $G^{(t-2)}$, $G^{(t-1)}$ and $G^{(t)}$ are generated and represents the equation (3):

$$G^{(t+1)} = f_{evolving}(r_{i,j}, G^{(t-2)}, G^{(t-1)}, G^{(t)}), \tag{3}$$

where $0 \leq t \leq R$, $0 \leq i, j \leq 8$ and R is a repeating counter for producing PRNS of a demanded cycle length.

Replace and updating phase. This phase replaces the *RNFv* and the *CPFv*. The method is represented as a following equation (4):

$$\begin{aligned} \text{new } RNFv &= G^{(t-2)} \oplus G^{(t-1)}, \\ \text{new } CPFv &= G^{(t-1)} \oplus G^{(t)} \end{aligned} \tag{4}$$

in $G^{(t+1)}$. The previous *RNFv* is replaced with a *new RNFv* and the previous *CPFv* is replaced with a *new CPFv*. Lastly, increase t by 1. If t is greater than R , finish the round. Otherwise, go back to the beginning of the evolving phase in 2-D.

4 Experimental Results

In this section, the quality of randomness of the 2-D V3-DCA PRNG is analyzed. The ENT [10] and the DIEHARD [9] test suites were utilized in order to analysis. The ENT test is useful for evaluating pseudorandom number generators for encryption and statistical sampling applications, compression algorithms, and other applications where the information density of a file is of interest [10]. The ENT test is a collective term for three tests, known as the Entropy test, Chi-square test, and Serial correlation coefficient (SCC) test. Normally before testing the quality of better with DIEHARD, it is first subjected to the ENT test.

The DIEHARD test suite is important because it appears to be the most powerful and difficult test suite to pass. This test consists of 18 different and

Table 1. The average values of the ENT and the result of DIEHARD test in p -value pass rate $\geq 85\%$

Test No.	Test Name	The average values				
		Shift register	Toma*	Guan ^A	Guan ^L	Proposed
1	Entropy (Closed to 8.0)	-	7.99971	7.99983	7.99815	7.99988
2	Chi-square (Closed to 1.0)	-	0.989412	0.992002	0.991591	0.999281
3	SCC (Closed to 0.0)	-	0.000227	0.000171	0.000216	0.000051
The results of the tests (Pass or Fail)						
4	Birthday Spacing	Pass	Pass	Pass	Pass	Pass
5	O5-P*	Pass	Fail	Fail	Fail	Pass
6	BR* 31 × 31	Fail	Pass	Pass	Pass	Pass
7	BR* 32 × 32	Fail	Pass	Pass	Pass	Pass
8	BR* 6 × 8	Fail	Pass	Pass	Pass	Pass
9	Bitstream	Pass	Fail	Pass	Fail	Pass
10	OPSO	Fail	Fail	Fail	Fail	Pass
11	OQSO	Fail	Fail	Pass	Pass	Pass
12	DNA	Pass	Fail	Pass	Pass	Pass
13	Count-The-1's 01	Fail	Fail	Pass	Pass	Pass
14	Count-The-1's 02	Fail	Pass	Fail	Fail	Pass
15	Parking Lot	Pass	Pass	Pass	Pass	Pass
16	Minimum Distance	Pass	Pass	Pass	Pass	Pass
17	3DS Spheres	Pass	Pass	Pass	Pass	Pass
18	Squeeze	Pass	Pass	Pass	Pass	Pass
19	Overlapping Sums	Pass	Pass	Pass	Pass	Pass
20	Runs	Fail	Fail	Fail	Pass	Pass
21	Craps	Pass	Pass	Pass	Pass	Pass

O5-P*: Overlapping 5-Permutation, BR*: Binary Rank,
 Toma*: Tomassini et al., Guan^A: Asymmetric neighborhood PRNG of Guan et al.,
 Guan^L: Lattice neighborhood PRNG of Guan et al.

independent statistical tests. The result of each test is called p -value. Most of the tests return a p -value in the DIEHARD, which should be uniform if the input file contains truly independent random bits. For any given test, a smaller p -value represents better results.

The proposed scheme produces a 64-bits output sequence at each round. The DIEHARD test suite requires a minimum of 10 MB regarding random number sequences [9]. Therefore, the proposed PRNG needs the $(10^7 \times 8) \div 64$ time rounds for the DIEHARD test. On the other hand, the ENT test suite requires fewer random number sequences, but the test is executed with the same 10 MB sequence for convenience and the next the DIEHARD test. A total of 100 experiments were performed. Table 1 shows the average value of the ENT test and the pass results of the DIEHARD for the 100 experiments. A pass is considered when all the p -values are passed by more than 65%, 75% and 85% at the 0.05 level. Table 2 shows that p -value pass rate more than 85% the 0.05 level. As the results show in Table, it has been proven that the quality of randomness of the proposed scheme is superior to other schemes.

5 Conclusion

In this paper, a new PRNG is proposed based on a virtual 3-D CA with the Moore neighborhood method. In order to achieve better randomness quality and decrease a correlation coefficient, the *RNF* and the *CPF* have been used. At the result, the various types of Moore neighborhood provides few correlation coefficients for each cell in state $G^{(t)}$, and a high randomness quality by the ENT and the DIEHARD test suites (average pass rate of more than 85% at the 0.05 level). Consequently, the proposed scheme preserves the quality of randomness in similarity to previous 2-D CA PRNGs and provides few area and time complexities (dose not using the genetic algorithm). Further more, the V3-DCA PRNG can produce randomness quickly, can be implemented conveniently by hardware, and can be used in many fields such as the built-in self-test of VLSI, keystream of One-time Pad cryptosystem, and symmetric key cryptography.

Acknowledgments

We would like to thank the anonymous reviewers for their helpful comments in improving our manuscript. This research was supported by the MKE (Ministry of Knowledge Economy) of Korea, under the ITRC support program supervised by the IITA (IITA-2008-C1090-0801-0026).

References

1. Tomassini, M., Sipper, M., Perrenoud, M.: On the generation of high quality random numbers by two-dimensional cellular automata. *IEEE Transactions on Computers* 49, 1146–1151 (2000)

2. Tomassini, M., Sipper, M., Zolla, M., Perrenoud, M.: Generating high-quality random numbers in parallel by cellular automata. *Future Generation Computer Systems* 16, 291–305 (1999)
3. Hortensius, P.D., Mcleod, R.D., Pries, W., Miller, D.M., Card, H.C.: Cellular automata-based pseudorandom number generators for built-in self-test. *IEEE Transaction Computer-Aided Design* 8, 842–859 (1989)
4. Wolfram, S.: Cryptography with cellular automata. In: Williams, H.C. (ed.) *CRYPTO 1985*. LNCS, vol. 218, pp. 429–432. Springer, Heidelberg (1986)
5. Guan, S.-U., Zhang, S., Quieta, M.T.: 2-D Variation With Asymmetric Neighborhood for Pseudorandom Number Generation. *IEEE Transaction on Computers* 23, 378–388 (2004)
6. Quieta, M.T.R., Guan, S.-U.: Optimization of 2D Lattice Cellular Automata for Pseudorandom Number Generation. *International Journal of Modern Physics C* 16(3), 479–500 (2005)
7. Tan, S.K., Guan, S.-U.: Evolving cellular automata to generate nonlinear sequences with desirable properties. *Applied Soft. Computing* 7, 1131–1134 (2007)
8. Chaudhuri, P.P., et al.: *Additive Cellular Automata Theory and Applications*, vol. 1, pp. 18–19. IEEE Computer Society Press, Los Alamitos (1997)
9. Diehard, G.M.: (1998), <http://www.stat.fsu.edu/pub/diehard>
10. ENT Test Suite, <http://www.fourmilab.ch/random>
11. Stallings, W.: *Cryptography and Network Security Principles and Practices*, 3rd edn., pp. 24–27. Prentice Hall, Englewood Cliffs (2003)
12. Winter, et al.: *Genetic Algorithms in Engineering and Computer Science*, pp. 24–27. Wiley Prentice Hall, New York (1995)
13. Kang, B.-H.: *A Pseudorandom Number Generator Based on Two-Dimensional Programmable Cellular Automata*, Doctor's Thesis, Kyungpook National University (June 2007)

An Adaptive Genetic Algorithm for Solving Traveling Salesman Problem

Jina Wang, Jian Huang, Shuqin Rao, Shaoe Xue, and Jian Yin

Department of Computer Science
Sun Yat-Sen University, Guangzhou, 510275
wangjina789@163.com

Abstract. Traveling salesman problem (TSP) is a classical NP-hard problem in combinational optimization. This paper adopted a novel genetic algorithm which adjust the crossover probability and mutation probability adaptively based on clustering and fuzzy system, and designed a new crossover operator to improve the performance of genetic algorithm (GA) for TSP. Experiments show that the proposed method is much better than the standard genetic algorithm with a higher convergent rate and success rate.

1 Introduction

Traveling salesman problem (TSP) is a classical NP-hard problem, and the model is widely applied in many practical optimized domains such as network and distribution optimization [1] [2]. The idea of TSP is to find a tour $\{C_{n(1)}, C_{n(2)}, \dots, C_{n(n)}\}$ from a given number of cities $C = \{C_1, C_2, \dots, C_n\}$ and the distance between each other $d(C_i, C_j)$, visit each city once and return to the starting city where the length of this tour is minimized. The objective function of TSP with n nodes is:

$$\min \sum_{i=1}^n d(C_{n(i)}, C_{n(i+1) \bmod(n)}) \quad (1)$$

The problem is equivalent to the Hamilton circle problem, which is a NPC problem [3]. Most of the researchers tend to adopt intelligence computation to achieve approximate optimized solution of the problem [4]. Genetic Algorithm (GA) is a powerful global randomized search and robust intelligent optimization technique based on natural evolution and derivative information. It has been used for solving complex problems such as NP-hard problems. Based on the observation that the parameters of GA decouple form the the whole GA process, some researchers propose that parameters modifying as a dynamic process that should be adaptively adjusted and integrated with the whole GA process.

In this paper, we propose a novel GA which adjusting the crossover probability P_c and mutation probability P_m adaptively based on clustering and fuzzy system [5]. We aim at avoiding SGA easily trapping into the local optimization or evolution halting states by corresponding fuzzy policy which adjusting the P_c and P_m by evaluating the maturity of the population. Considering the sequential characteristic of TSP, some

widely used crossover operators [6] destroy the valid patterns of GA when they guarantee the validity of TSP. We design a novel crossover operator, and adopt a one-point crossover to make the operator work effectively. Experiments show that the proposed method is much better than SGA with high convergent rate and success rate.

The article is organized as follows: Section 2 gives an introduction of SGA process and presents the proposed adaptive GA in detail; Section 3 gives the special operators of TSP; Section 4 provides a number of experiment results; and finally concludes will be given in section 5.

2 Genetic Algorithm and an Adaptive Genetic Algorithm Based on Clustering

The idea of GA is first proposed in 1975 and it is an intelligence algorithm which simulates the behavior of the biology evolution [6] [7]. SGA process consist the following steps: encoding, selection, crossover, mutation and decoding. GA begins with the chromosomes which have been encoded to represent a group of vectors in the solution space, and adopts the similarity with the optimized solution as the fitness after initialize the population, then selects the parent individuals for the next generation according to the fitness, after that generates the next generation recur to the crossover operator and mutation operator of the natural genetics, finally executes the previous process iteratively to make the whole population evaluate towards the direction close to the optimized solution. The decoded vector of the final chromosomes can be viewed as an approximate optimized solution of the problem. When generate the next generation, P_c and P_m affect the process greatly. The bigger the P_c is, the more abroad the mutual information communication among the population, the bigger the P_m is, the faster the generation and the complementarily for new patterns. Traditional GA operators didn't consider the changing of the information communication and the pattern complementarities during different evolution phases [8] [9].

This section we will adjust P_c and P_m based on the evolution phase to make the population evolve towards the overall fitness increasing direction. The outline of the adaptive adjusting GA we used is described in Tab.1.

2.1 Maturity Evaluation Based on Clustering

We adopt K-means clustering algorithm to evaluate the population maturity. By applying the K-means algorithm, the distribution of the population in the search space is clustered in each generation. K-means clustering algorithm can be viewed as a classify algorithm which partition the original population space into some sub-population space with closed spatial positions according to the individuals' fitness or the position adjacency of feature space, and the sub-population don't intersect with each other. The adaptive adjusting GA cluster in the solution space through distances between each individual. It partitions the chromosome population in the solution space into several clusters with minimum covariance, and evaluate the maturity according to the size of the clusters and whether the cluster contains the best or the worst individual. The clustering process is as follows: 1) Initialize cluster number k and find a set of centers

Table 1. Outline of the Adaptive Adjusting Genetic Algorithm for TSP

step 1. Initialize the population by random generating a tour $\{C_{n(1)}, C_{n(2)}, \dots, C_{n(n)}\}$.
 step 2. Evaluate the fitness.
 step 3. Roulette wheel selection which is stated in [10].
 step 4. Adjusting P_c and P_m :1) Maturity evaluation based on clustering. 2) Fuzzy system calculate δP_c and δP_m .3)Obtain P_c and P_m .
 step 5. Cross according to P_c .Mutate according to P_m .
 step 7. Evaluate the fitness for each individual in the population.
 step 8. Go to step 3 and repeat until convergence.

$\mu = \{\mu_1, \mu_2, \dots, \mu_k\}$ from the population vectors $C = \{C_1, C_2, \dots, C_n\}$; 2) Assign C_i ($i \in \{1, \dots, n\}$) to a cluster which the distance between it and the cluster center is the shortest; 3) Update the new cluster centers with the new sub-population belong to it; 4) Go to 2), iteratively run until convergence.

2.2 Calculation for P_c and P_m Based on Fuzzy Mechanism

The calculation for P_c , P_m and the adjusting values of them is based on considering the relative size of the cluster which contains the best chromosome and the one which contains the worst chromosome, note the sizes of them as G_B and G_W respectively. After the maturity evaluation based on clustering, we can obtain the adjusting direction of P_c and P_m according to the four rules defined in [5]. The definition of increase or decrease direction is described in Tab.2.

Table 2. Rules for Adjusting P_c and P_m

Rule 1) Reduce P_c and P_m if G_B is the largest and G_W is the smallest among the clusters;
 Rule 2) Increase P_c and reduce P_m if G_B equals G_W and they are the largest compared with others;
 Rule 3) Increase P_c and P_m if G_B equals G_W and they are the smallest compared with others;
 Rule 4) Reduce P_c and increase P_m if G_B is the smallest and G_W is the largest in comparison to the others.

In order to simplify the calculation, we first normalize G_B and G_W to a interval ranging from zero to one, then calculate the relationship between G_B and the big fuzzy set P_B , G_W and the small fuzzy set P_S , so as to reflect the size respectively. Finally, gain the adjusting range δP_c and δP_m respectively of the P_c and the P_m by applying the “center of sum method”. In addition, in order to assure the continuous of the adjusting process, we have to add a limitation to the adjusting range in each generation, which means that the increase or decrease of P_c and P_m shouldn’t exceed a proportion compared with themselves respectively. The proportion can be different according to the problem. The calculated equations are described as Eq.(2) and Eq.(3), K_c and K_m are the suitable adjusting proportions which are chosen to keep the changes of the values of P_c and P_m within a tolerance limitation in each generation.

$$P_c(gen) = P_c(gen - 1) + K_c \delta P_c(gen) \tag{2}$$

$$P_m(\text{gen}) = P_m(\text{gen} - 1) + K_m \delta P_m(\text{gen}) \quad (3)$$

3 Special Operators of Traveling Salesman Problem

3.1 Crossover Operator and Mutation Operator

In order to assure that child chromosome as a sequence and the valid pattern derived from the parent chromosomes not be disordered excessively, we design a novel crossover operator for TSP according to the following criteria: Do not arouse the sequence adjusting out of the crossover segment while adjusting the crossover segment to a valid sequence, consequently, the inherited valid pattern wouldn't destroyed by constructing valid pattern.

When mostly guarantee the valid pattern of the outer crossover segment by constructing the valid solution according to the criteria stated above, the two individuals can't crossover sufficiently if the crossover segment is too short due to the strong constraint. So we have to consider the average crossover length carefully when we decide the selection for the crossover segment. The longer the average length, the more sufficient the crossover between two individuals.

The crossover average length of one-point is $\sum_{i=1}^{n-1} i/(n-1)$, where n is the length of the gene, the result is $n/2$, namely $1/2$ of the gene length. The crossover average length of two-point is $\int \int_{1 \leq x, y \leq n-1} |y-x| dx dy / (n-1)$. The result is $n/3$, namely $1/3$ of the gene length.

Based on the analysis above, we adopt the one-point crossover, the crossover detail is described as follows: 1) Identify the common component of the two crossover chromosomes; 2) Exchange elements in the crossover segment, and use the identity to record the duplication of the two sub-chromosomes in the exchange process; 3) Re-exchange the duplication elements to the original sequence. For example, $A: 458936271$ and $B: 168723945$ are two gens for crossover operation, suppose we crossover from the fourth, so the sub-genes for crossing are 936271 and 723945 . We first identify that the common component is 9327 , then exchange the two sub-genes, it is easily to show that the two elements of 45 and 61 are duplicated for gene A and gene B respectively, so we re-exchange them to the original sequence, which means that substitute 45 with 61 , and substitute 61 with 45 , then the new constructed genes are 458723961 and 168934275 . It is easily to see that the two sequences greatly maintain the sequence pattern derived from parents A and B while assuring the feasibility. We construct the mutation operator based on the random exchanging, which random select two positions.

3.2 Pseudo-random Sequence Generation Operator

We have to initialize the chromosome as a random sequence in the first step of GA. It is time costly for the long random sequence construction because the production of random sequence by the general construction set and the mutex check policy is proportional to the square of the average sequence length. We adopt a pseudo-random sequence generation algorithm based on random exchanging, and the pseudo-random sequence is certainly approximate to the complete random sequence.

The construction of pseudo-random sequence is described as follows: 1) Assign to each position of the sequence according to the natural number sequence; 2) Generate a random number ranging from 1 to the sequence length for the element in the first position of the sequence, and exchange the elements in the first position and the random position. Note that the first element may not be changed because the random position may be 1; 3) Take the rest of the sequence to exchange randomly like step 2).

4 Experiment

The adaptive GA for TSP was implemented in Matlab on an Intel Pentium D 3.00 GHz and 1.0 G memory personal computer under the windows operating system. We adopt integer encoding and set the population size $N_p = 103$, the adaptive adjusting parameter $K_c = 0.015$, $K_m = 0.0015$, crossover probability $P_c = 0.75$, mutation probability $P_m = 0.01$ in the initial step, and limit $P_c \in [0.60, 0.95]$ and $P_m \in [0.005, 0.05]$. We adopt the inverse of the total sequence length as the fitness value in the problem domain, the larger the fitness, the shorter the tour, which means the optimum solution for the TSP problem.

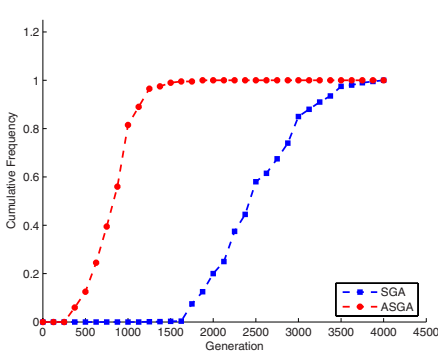
Virtual Dataset. We set the number of nodes range from 50 to 200, and random generate the distances which satisfy the uniform distribution in the interval [1000, 10000]. The comparison for the cumulative frequency between the Self-Adaptive GA algorithm(SAGA) and the standard GA (SGA) with different number of nodes and iterative times can be seen in Tab.3.

Table 3. Comparison for the Cumulative Frequency between SGA and SAGA

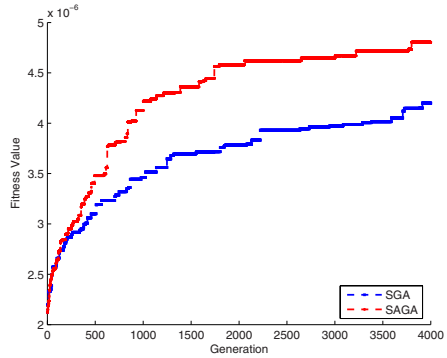
Iterative Times	50 nodes		100 nodes		150 nodes		200 nodes	
	SGA	SAGA	SGA	SAGA	SGA	SAGA	SGA	SAGA
500	0	0	0	0	0	0	0	0
1000	0	0.015	0	0.395	0	0.005	0	0
1500	0	0.12	0.025	0.965	0	0.11	0	0.18
2000	0.005	0.335	0.20	1	0	0.645	0	0.89
2500	0.015	0.525	0.58	1	0	0.965	0.01	1
3000	0.015	0.71	0.85	1	0.03	1	0.15	1
3500	0.06	0.81	0.975	1	0.15	1	0.39	1
4000	0.08	0.885	1	1	0.38	1	0.695	1

From Tab.3 it is not difficult to see that the cumulative frequency is smaller for some small size of TSP, this is because that the success standard we choose is given by a relative value to a certain problem, but not the optimal solution. This is not prejudice against the comparison for the two algorithms. We can see from Tab.3 that for the same size TSP problem, SAGA is in much higher convergent rate than SGA, and the phenomenon is even more significant with the increasing of the size.

For 100 nodes traveling salesman problem, Fig.1(a) shows the cumulative frequency comparison between standard GA and the adaptive adjusting GA along with the increasing iterative times. Fig.1(b) is the fitness value comparison between SGA and SAGA



(a) Cumulative Frequency Comparison



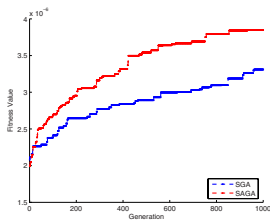
(b) Fitness Value Comparison

Fig. 1. For 100 nodes traveling salesman problem, with the increasing of iterative times, the comparison of SGA and SAGA algorithm on the cumulative frequency and the fitness

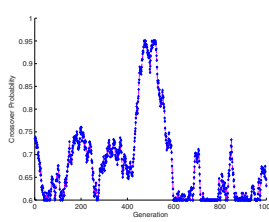
along with the increasing iterative times. It is clearly that the cumulative frequency and the average fitness value of SAGA are much higher than that of SGA when in the same iterative times. In fact, the fitness value gap between SGA and SAGA is much wider in practical because we use the inverse of the total sequence length as the fitness value, which narrow the fitness gap mostly.

Fig.2(b) shows the P_c curve for 1000 generation with 100 nodes. There are best individuals and worst individuals during every optimization phases, so we can see that it increase and decrease by turns, but the scope is different. The main trend of P_c is decreasing in the initial and matured states which would like to mostly reduce the properties of worst chromosomes and maintain the properties of best chromosomes respectively, and increasing in the sub-maturing and maturing states which aim to explore new search directions for enhancing the growth of the best candidates. Fig.2(c) is the P_m curve for 1000 generation, we can see that it satisfies the optimization strategies.

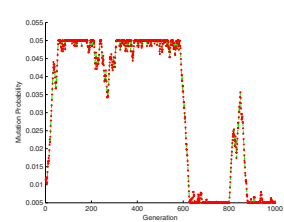
TSPLIB Dataset. Our algorithm have also tested on TSPLIB, the results show that SAGA is sound. SAGA best tour is close to the optimal tour provided by TSPLIB within



(a) Fitness Value Comparison



(b) Crossover Probability



(c) Mutation Probability

Fig. 2. For 1000 generation with 100 nodes, Fig.2(a) shows the fitness comparison of SGA and SAGA, Fig.2(b) and Fig.2(c) represent the P_c and P_m changing courses respectively

Table 4. TSPLIB Problems Solved by SAGA and SGA Within a Certain Generation

Problem	Problem Size	Optimal Tour by TSPLIB	Generation	SAGA Best Tour	SGA Best Tour
<i>br17</i>	17	39	1000	39	55
<i>ftv33</i>	34	1286	4000	1453	1895
<i>ftv55</i>	56	1608	4000	1672	1884
<i>p43</i>	43	5620	2500	5783	11128
<i>ry48p</i>	48	14422	4000	14531	15024
<i>ft53</i>	53	6905	4000	6919	7032
<i>ft70</i>	70	38673	4000	38912	41056
<i>kro124</i>	100	36230	4000	36312	36406

a certain generation, while SGA can't give a satisfying solution, as shown in Tab.4. For example, the best tour of SAGA is 5783 within 2500 generation for the *p43* dataset whose optimal tour is 5620, while the SGA best tour is 11128 within 2500 generation, which is far from the optimal tour. For other datasets, SAGA provides a best solution the same as or close to the optimal solution given by TSPLIB, thus the effectiveness and convergence of SGA are lower.

5 Conclusion

In this paper, an adaptive adjusting of P_c and P_m based on clustering and fuzzy system was developed for the TSP, and we designed a new crossover operator to improve the performance of GA. Clustering technique was used to evaluate the maturity of the population, and fuzzy mechanism was used to calculate the change δP_c and δP_m for P_c and P_m respectively. The comparisons show that our method is sound.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (60573097, 60773198, 60703111), Natural Science Foundation of Guangdong Province (05200302, 06104916), Specialized Research Fund for the Doctoral Program of Higher Education(20050558017), Program for New Century Excellent Talents in University of China (NCET-06-0727).

References

1. Cai, H.Z., Hu, J.G., Gao, W., Wei, W., Tang, L.S.: An Improved Evolutionary Algorithm for the Traveling Salesman Problem. Chinese Journal of Computers 28, 823–828 (2005)
2. Pullan, W.: Adapting the genetic algorithm to the travelling salesman problem. IEEE Transactions on Evolutionary Computation 2, 1029–1035 (2003)
3. Dorigo, M., Gambardella, L.M.: Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Transactions on Evolutionary Computation 1, 53–66 (1997)
4. Johnson, D.S.: Local Optimization and the Traveling Salesman Problem. In: Paterson, M. (ed.) ICALP 1990. LNCS, vol. 443, pp. 446–461. Springer, Heidelberg (1990)
5. Zhang, J., Chung, H., Lo, W.L.: Clustering-Based Adaptive Crossover and Mutation Probabilities for Genetic Algorithms. IEEE Transactions on Evolutionary Computation 11, 326–335 (2007)

6. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley, Reading (1989)
7. De Jong, K.A.: An analysis of the behavior of a class of genetic adaptive systems. Ph.D. dissertation, Univ. Michigan, Ann Arbor, MI (1975)
8. Yang, S.X.: Adaptive Crossover in Genetic Algorithms Based On Statistics Mechanism. In: Proceedings of the eighth international conference on Artificial life, pp. 182–185 (2002)
9. Yang, S.X.: Adaptive Non-uniform Crossover Based On Statistics For Genetic Algorithms. In: Proceedings of the eighth international conference on Artificial life, pp. 182–185 (2002)
10. De Michelis, M.: Genetic Algorithms+Data Structures = Evolution Programs. Springer, Berlin (1996)
11. Srinivas, M., Patnaik, L.: Adaptive probabilities of crossover and mutation in genetic algorithms. *IEEE Transactions on Systems, Man and Cybernetics* 24, 656–667 (1994)

An Application of Support Vector Machines for Induction Motor Fault Diagnosis with Using Genetic Algorithm

Ngoc-Tu Nguyen and Hong-Hee Lee

School of Electrical Engineering, University of Ulsan, Ulsan, South Korea
nntu@hcmut.edu.vn, hhlee@mail.ulsan.ac.kr

Abstract. This paper introduces a technique for diagnosing mechanical faults of induction motors by using support vector machine (SVM) and genetic algorithm (GA). Features are extracted from the vibration time signals and selected by using GA with a distance evaluation fitness function. All SVM parameters are also obtained simultaneously by the same GA. The SVM is studied with two types of kernel functions, the radial basis function and the polynomial function. Four motor conditions are investigated with the chosen SVM classifiers. The classification results have high accuracy for the chosen feature set and SVM parameters.

Keywords: Induction motor faults; diagnosis; support vector machine; genetic algorithm; distance criterion.

1 Introduction

In these days, a fault diagnostic technology for the rotating machine, such as induction motors, continues to grow rapidly. When the fault occurs, it affects machine dynamic conditions as vibration, sound, temperature, etc. and they can be useful condition indicators. Based on these symptoms, many types of diagnostic methods such as support vector machine (SVM) [1-6], adaptive neuro-fuzzy inference system [7-8], decision tree [9], artificial neural network (ANN) [10], and nearest neighbors' rule [11], etc. have been developed. Of all these methods, the SVM is a supervised learning method used for classification and regression with a high accuracy and a good generalization capability. In [1], SVM is applied to perform the fault identification with the input data extracted by the principal component analysis (PCA) and the independent component analysis (ICA). But its SVM parameters are selected by trials. In [4], the SVM parameters are chosen by the averaged value of the standard deviation of the data, but it is purely an empirical method. In [5], the authors have used GA technique for feature selection and the kernel parameters are applied to the bearing fault diagnosis.

To acquire the fault information, some statistical characteristics are calculated from vibration signals to form the feature set. But, the data are still possible to contain some irrelevant or redundant features as well as useful features in this feature set. If

all features are used as input of a classifier, the classification processes become slow and the classification performance can be deteriorated. Basically, there are many methods to overcome these problems, and they can be the ICA, the distance criterion feature selection [1], the PCA method [2, 9], the feature selection using the decision tree [6], etc. The data transformation to a lower dimension of ICA and PCA techniques is effective, but there is no guarantee that the removed information is not essential for fault classification. In specific cases, the feature selection that selects a part of original data can give better outcomes.

In this paper, we propose the SVM classifier which can select optimal parameters and feature subset simultaneously using the GA; GA technique is used for not only selecting the SVM features and all SVM parameters but also applying distance evaluation technique for the fitness function. Also, this paper expands the diagnostic classes to deal with the mechanical faults of induction motors: bearing damage, looseness, and rotor unbalance. The motor conditions are monitored and diagnosed using the proposed SVM, and the results are listed with high performances in the experimental section.

2 Support Vector Machines

For illustration, a typical two-class classification is given as an example that includes positive and negative classes. With a given data set $\{(x_i, y_i), i = 1, 2, \dots, n\}$ where $x_i \in R^n$ and y_i is either 1 or -1, the SVM operates by finding a hyper-plane in the space of possible inputs. This hyper-plane will try to split the positive samples from the negative samples. The separation is chosen to have the largest distance from the hyper-plane to the nearest of the positive and negative samples. The dividing hyper-plane can be expressed as the form:

$$\omega \cdot x + b = 0 \quad (1)$$

where ω is the weight vector which is perpendicular to the separating hyper-plane. The bias b allows extending the margin. In case of the linear separation, the data points are classified by

$$\omega \cdot x + b \geq 1 \quad \text{for all } y_i = 1 \quad (2)$$

$$\omega \cdot x + b \leq -1 \quad \text{for all } y_i = -1 \quad (3)$$

The nearest samples that are used to define the margin are called the support vectors. The support vectors contain all information that is needed to define the classifier. In order to get the optimal separating hyperplanes or to maximize the margin $2/|\omega|$ between two classes, the problem becomes solving a quadratic programming optimization problem to minimize $|\omega|$:

$$\begin{aligned} \text{Minimize} \quad & (1/2) \|\omega\|^2 + C \sum_i \xi_i \\ \text{Subject to} \quad & y_i(\omega \cdot x + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, 1 \leq i \leq n, \end{aligned} \quad (4)$$

where ξ_i is slack variables which measure the degree of misclassification of the data x_i and C is error penalty constant.

The equation (4) is solved to get the optimal solutions of \mathbf{w}^* and \mathbf{b}^* . Then, the discrimination function is obtained as (5).

$$f(x) = \text{sgn}(\mathbf{w}^* \cdot \mathbf{x} + b^*) \quad (5)$$

In case the linear boundary in the input spaces is not enough to separate into two classes properly, a nonlinear classification is suggested. In nonlinear classification, the SVM maps the data from input space to feature space by using the kernel function which can produce the maximum margin hyperplanes. The basis form of the discrimination function now can be shown as

$$f(x) = \text{sgn}\left(\sum_{i=1}^n y_i a_i k(x_i, x_j) + b\right) \quad (6)$$

Some common kernel functions are as follows

Polynomial kernel:

$$k(x_i, x_j) = (\gamma \cdot x_i^T x_j + r)^d, \gamma > 0 \quad (7)$$

Radial basis function kernel:

$$k(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0 \quad (8)$$

Gaussian radial basis function kernel:

$$k(x_i, x_j) = \exp\left(-\|x_i - x_j\|^2 / (2\sigma^2)\right) \quad (9)$$

Sigmoid kernel:

$$k(x_i, x_j) = \tanh(\gamma \cdot x_i^T x_j + r) \quad (10)$$

Where, γ , r , d , σ are kernel parameters.

The kernel function such can decide the complexity of classification function set. More detail about SVM can be found in [13-15].

3 Experimental Setup and Feature Extraction

A tri-axial accelerometer is mounted at the housing near the bearing to measure vibration signals. The vibration signals are sampled by a data recorder. Time-domain vibration signals are extracted into 18 features. Among these features, each 6-feature set is constructed from one of three different signals which are measured in three directions: axial, horizontal, and vertical. The features are formed as follows: root mean square(a), variance(a), skewness(a), kurtosis(a), crest factor(a), maximum(a), root mean square(h), variance(h), skewness(h), kurtosis(h), crest factor(h), maximum(h), root mean square(v), variance(v), skewness(v), kurtosis(v), crest factor(v),

maximum(v). The terms (a), (h), and (v) respectively represent axial, horizontal, and vertical. These features represent the energy, the vibration amplitude, and the time series distribution of the signal in time-domain.

To induce failure in the induction motor, bearings with cage and ball damage are used. In addition, rotor unbalance is simulated by creating a hole in the rotor and adding a given mass. A loose bearing motor from the factory is used to simulate the looseness fault.

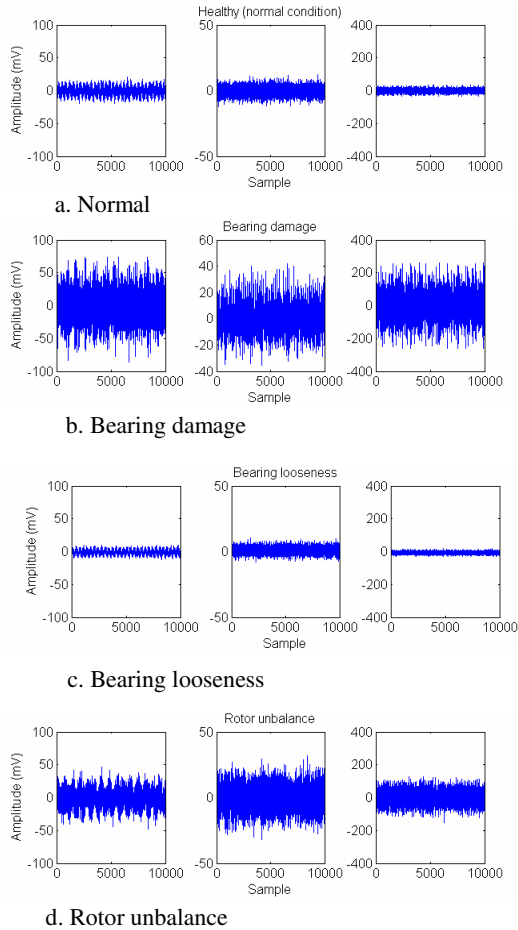


Fig. 1. Time-domain vibration signals in three directions: (a) Normal (b) Bearing damage (c) Bearing looseness (d) Rotor unbalance (from left to right: horizontal, axial, and vertical direction)

Fig. 1 shows sample signals of four motor conditions are measured in 3 directions x, y, and z. These data are preprocessed by using the equations in Table 1 to form 18-feature training and test set. Each feature is normalized by dividing it by the maximum of its absolute value before being used for training and testing the SVM models.

Table 1. Time-domain features

Feature	Equation
Root mean square	$rms = \sqrt{\frac{\sum_{n=1}^N (x(n))^2}{N}}$
Variance	$var = \sigma^2 = \frac{\sum_{n=1}^N (x(n) - mean(x))^2}{(N-1)}$
Skewness	$skewness = \frac{\sum_{n=1}^N (x(n) - mean(x))^3}{(N-1)\sigma^3}$
Kurtosis	$kurtosis = \frac{\sum_{n=1}^N (x(n) - mean(x))^4}{(N-1)\sigma^4}$
Crest factor	$crest = \frac{\max x(n) }{rms}$
Maximum value	$max = \max x(n) $

4 Genetic Algorithm

A genetic algorithm (GA) is a search technique used for finding out true or approximate solutions of the optimization and the search problems. GA is used to give the solution by simulating the evolutionary processes of survival of the fittest which ensures that the best members of population are retained. GA generates successive populations of alternate solutions by using techniques such as inheritance, selection, mutation, and crossover until the acceptable results are obtained. The algorithm begins with a set of solutions (chromosomes) called population. In each generation, the fitness of every chromosome is evaluated, and then some chromosomes are selected based on their fitness and modified to reproduce a new population. The modification stage includes the selecting pairs of parents, the crossover of genes to produce new ones, and the mutation to randomly alter the genes. The crossover exchanges genes between two chromosomes, and then mutation alters genes code from 1 to 0 or vice versa in case of binary genes code. Last, an optimal solution can be obtained after a series of iterative computations.

There are many random procedures which are called during execution of GA algorithm. Therefore, the randomness plays a central role in GA processes. As a result, GA may have a tendency to converge towards a local optimal rather than a global optimum of the problem. But in general, GA can rapidly find out good solutions even in the difficult search spaces.

4.1 Chromosome

In this paper, the SVM with RBF (radial basis function) and polynomial kernel functions is used for the fault classification. GA is designed for the feature selection and the SVM parameter optimization. For this purpose, the chromosome has been formatted in a bit string form, where the first 18 bits represent 18 features, and next 8 bits represent C and γ parameters. In case of the polynomial kernel function, there are additional two parameters, d and r, as shown in Fig. 2b. The value of d is chosen between 1 and 4, represented by 2-bit binary code.

F1	F18	C1 ... C4	G1 ... G4
----	-------	-----	-----------	-----------

a. RBF chromosome format

F1	F18	C1 ... C4	G1 ... G4	D1 D2	R1 ... R4
----	-------	-----	-----------	-----------	-------	-----------

b. Polynomial chromosome format

Fig. 2. The chromosome formats

In Fig. 2, the terms F1- F18 represent the features, in which the value ‘1’ means the feature is selected and the value ‘0’ indicates feature is not selected. The terms C1 - C4 represent the value of constant C, and the terms G1- G4, D1-D2, and R1-R4 represent the parameters γ , d and r, respectively. The real values of C, γ , d, and r parameter can be decoded from their binary codes using the following equation:

$$\text{ParamValue} = VLB + \frac{VUB - VLB}{2^n - 1} d \tag{11}$$

where d is decimal value of bit string,
 VLB is lower boundary value of the parameter,
 VUB is upper boundary value of the parameter,
 n is length of binary code.

4.2 Fitness Function

Fitness function is an important factor in a point of the speed and the efficiency of the algorithm. In this paper, the fitness function of GA is developed on the basis of the SVM training accuracy and the number of selected features. The SVM accuracy is obtained by the evaluation of the test data classification using the trained model. By using this fitness function, SVM parameters are optimized and the number of feature is also selected. The GA chooses the chromosome with the smallest fitness value after finishing the last iteration.

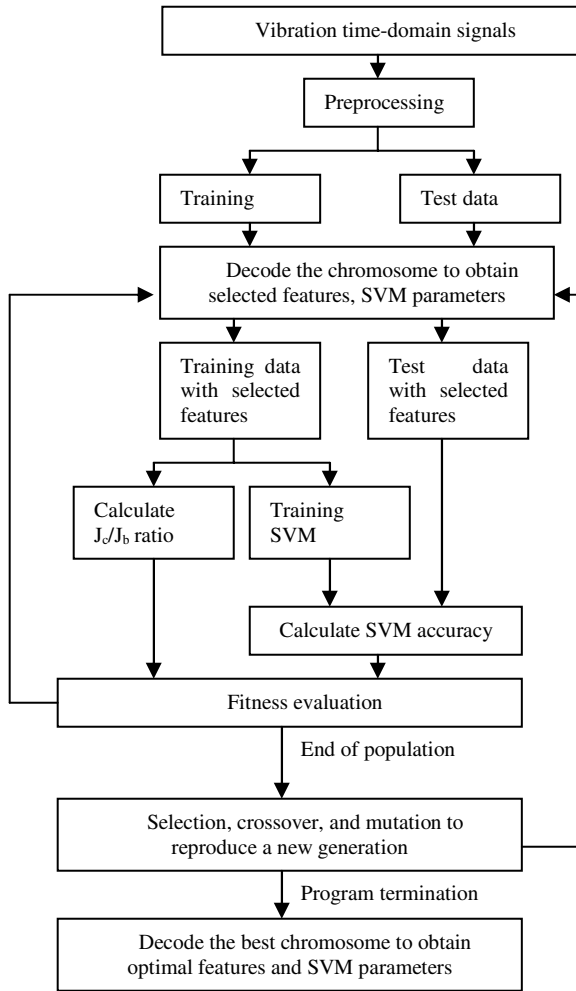


Fig. 3. The flow chart of the system

The fitness function of GA is formed as follows

$$F = W * (SVM_training_accuracy)^{-1} + (1-W) * \frac{\sum_{classes} \text{within-class distance}}{\sum_{classes} \text{between-class distance}} \quad (12)$$

Equation (12) can be rewritten in (13):

$$F = W * (SVM_training_accuracy)^{-1} + (1-W) * \frac{J_c}{J_b} \quad (13)$$

where W is weighting factor with the value between 0.0 and 1.0.

The within-class distance is given by

$$J_c = \sum_{i=1}^c p_i J_i \quad (14)$$

$$J_i = (1/n_i) \sum_{k=1}^{n_i} (x_k^i - m_i)^T (x_k^i - m_i) \quad (15)$$

where class $i = 1, \dots, c$; m_i is the mean vector of class i ; n_i is the number of samples in class i ; p_i is the number of samples in class i .

The between-class distance is

$$J_b = \sum_{i=1}^c p_i (m_i - m)^T (m_i - m) \quad (16)$$

where m is the mean vector of all of the classes.

The ratio J_c/J_b is used to obtain the optimal features based on the criterion that chooses the smaller within-class distance J_c and the larger between-class distance J_b . The number of features is selected according to the smaller J_c/J_b .

4.3 Block Diagram

Fig. 3 presents a block diagram of GA program that is used to look for an optimal feature subset and SVM parameters. Each member in population is evaluated by its fitness value, and then the best members are selected to reproduce a new population. When the termination condition is satisfied, the best member is obtained and decoded to receive the needed information.

5 Experimental Results

The training data includes 559 samples and the test data which have 307 samples are used in this paper. The kernel functions of SVM are the RBF and the polynomial function. The GA parameters are: VLB = 0.01, VUB = 50, number of iterations = 200, mutation rate = 0.7, 1 crossing point crossover, population size = 96. The weighting factor W is altered to get different set of features, and changed from 0.6 to 0.9. Results are listed in Table 2 and 3 for two cases, RBF and the polynomial kernel function, respectively.

The selection with smaller number of features and the larger accuracy can be chosen from the classification performances and the feature sets that are listed in Table 2 and 3. For the SVM with RBF in Table 2, the feature set (1, 9, 10, 13, and 18) seems to be the best choice. This set has only 5 features but gives the highest performance compared with the others. In Table 3, when the polynomial kernel function is used, the set (6, 9, 10, and 12) can be the best one. It has the smallest number of features (4 features) and a rather high performance. Comparisons with the full features SVM models are shown in Table 4. In order to confirm the efficiency of the proposed selections, Table 4 shows the performances of the SVM for the selected feature subsets and

Table 2. Experimental results using GA with RBF kernel function

Selected features	Parameter C	Parameter γ	Training accuracy (%)	Test data accuracy (%)	Normal (%)	Looseness (%)	Bearing damage (%)	Unbalance (%)
1, 2, 3, 9	40.0020	46.6673	95.89	93.49	100	74.24	100	96.81
1, 9, 10, 13, 18	30.0040	46.6673	97.50	99.35	100	96.97	100	100
1, 3, 9, 13, 14	13.3407	33.3367	95.17	93.81	100	75.76	100	96.81
1, 2, 3, 9, 13	43.3347	10.0080	93.92	97.07	100	87.88	100	98.94
1, 3, 9, 10, 11, 13	46.6673	26.6713	98.75	91.53	100	74.24	100	90.43
1, 3, 9, 10, 13, 14, 15, 16, 18	43.3347	13.3407	99.82	93.16	98.21	78.79	100	93.62

Table 3. Experimental results using GA with polynomial kernel function

Selected features	Parameters (C, γ , r, d)	Training accuracy (%)	Test data accuracy (%)	Normal (%)	Looseness (%)	Bearing damage (%)	Unbalance (%)
6, 9, 10, 12	23.3387, 6.6753, 3.3427, 4	99.11	97.39	100	100	93.41	97.87
1, 2, 6, 10, 18	30.004, 13.3407, 43.335, 3	99.28	97.39	100	100	100	91.49
1, 6, 9, 10, 13, 16	43.3347, 23.3387, 0.01, 1	98.21	95.44	100	100	86.81	97.87
1, 6, 9, 10, 11, 13, 16, 18	33.3367, 43.3347, 6.6753, 1	98.21	96.09	100	100	89.01	97.87
1, 6, 7, 9, 10, 1, 2, 16, 18	23.3387, 40.002, 50, 3	100	94.46	100	100	100	81.91
1, 2, 3, 6, 9, 10, 12, 13, 16, 18	46.6673, 0.01, 3.3427, 2	96.96	99.02	100	100	100	96.81

all feature set. The all feature set experiments are repeated 100 times with random parameters to get the average results. The SVM's parameters are chosen randomly in the range from 0 to 50.

The experimental results show the efficiency of the proposed selections for both data dimension and SVM parameters compared to the efficiency without feature selection. The accuracy is improved with the proposed models while data dimension is decreased significantly. For RBF case, the accuracy is increased from 94.61% to 99.35% with 72.2% data reduction. In case of the polynomial kernel function, the accuracy is increased to 97.39% with 77.8% data reduction. Generally, Table 4 shows that SVM have high accuracy and performance for motor fault diagnosis in this work.

Table 4. Performance of proposed SVM model compare to full data SVM model

Kernel function and selected features	Parameters	Training accuracy (%)	Test data accuracy (%)	Normal (%)	Loose-ness (%)	Bearing damage (%)	Unbalance (%)
RBF (all features)	-	97.30	94.61	90.07	95.03	97	94.71
Polynomial (all features)	-	90.34	85.47	79.86	62.70	95	95.59
RBF (1, 9, 10, 13, 18)	30.0040, 46.6673	97.50	99.35	100	96.97	100	100
Polynomial (6, 9,10,12)	23.3387, 3.3427, 4, 6.6753,	99.11	97.39	100	100	93.41	97.87

6 Conclusion

In this paper, the vibration data are processed to get high accuracy SVM for both RBF and polynomial kernel functions. The trained SVM are applied to the fault diagnosis of induction motors. According to the experimental results, the SVM classification method is proved as an efficient way for fault diagnosis of induction motors.

GA algorithm is used in this paper to find out the proper SVM parameters and feature subset simultaneously. It shows the importance of selecting the parameters and training inputs for SVM that have big influence on the system performance. The feature selection can remove the irrelevant and the redundant information by choosing useful features as input of SVM, while its proper parameters help to build SVM model with high performance and accuracy.

Acknowledgments. This work was supported by the Research Fund of University of Ulsan.

References

1. Widodo, A., Yang, B.S., Han, T.: Combination of Independent Component Analysis and Support Vector Machines for Intelligent Faults Diagnosis of Induction Motors. *Expert Systems with Applications* 32, 299–312 (2007)
2. Yuan, S.F., Chu, F.L.: Fault Diagnostics Based on Particle Swarm Optimization and Support Vector Machines. *Mechanical Systems and Signal Processing* 21, 1787–1798 (2007)
3. Yang, Y., Yu, D., Cheng, J.: A Fault Diagnosis Approach for Roller Bearing Based on IMF Envelope Spectrum and SVM. *Measurement* 40, 943–950 (2007)
4. Jack, L.B., Nandi, A.K.: Fault Detection Using Support Vector Machines and Artificial Neural Networks, Augmented by Genetic Algorithms. *Mechanical Systems and Signal Processing* 16, 373–390 (2002)
5. Samanta, B., Al-Balushi, K.R., Al-Araimi, S.A.: Artificial Neural Networks and Support Vector Machines with Genetic Algorithm for Bearing Fault Detection. *Engineering Applications of Artificial Intelligence* 16, 657–665 (2003)

6. Sugumaran, V., Muralidharan, V., Ramachandran, K.I.: Feature Selection Using Decision Tree and Classification through Proximal Support Vector Machine for Fault Diagnostics of Roller Bearing. *Mechanical Systems and Signal Processing* 21, 930–942 (2007)
7. Lei, Y., He, Z., Zi, Y., Hu, Q.: Fault Diagnosis of Rotating Machinery Based on Multiple ANFIS Combination with Gas. *Mechanical Systems and Signal Processing* 21, 2280–2294 (2007)
8. Ye, Z., Sadeghian, A., Wu, B.: Mechanical Fault Diagnostics for Induction Motor with Variable Speed Drives Using Adaptive Neuro-fuzzy Inference System. *Electric Power Systems Research* 76, 742–752 (2006)
9. Sun, W., Chen, J., Li, J.: Decision Tree and PCA-based Fault Diagnosis of Rotating Machinery. *Mechanical Systems and Signal Processing* 21, 1300–1317 (2007)
10. Samanta, B., Al-Balushi, K.R., Al-Araimi, S.A.: Artificial Neural Networks and Genetic Algorithm for Bearing Fault Detection. *Soft Comput.* 10, 264–271 (2006)
11. Casimir, R., Boutleux, E., Clerc, G., Yahoui, A.: The Use of Features Selection and Nearest Neighbors Rule for Faults Diagnostic in Induction Motors. *Engineering Applications of Artificial Intelligence* 19, 169–177 (2006)
12. Rao, J.S.: *Vibratory Condition Monitoring of Machines*, pp. 361–382. Alpha Science International Ltd, UK (2000)
13. Chang, C.C., Lin, C.J.: LIBSVM: A Library for Support Vector Machines, <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
14. Hsu, C.W., Chang, C.C., Lin, C.J.: *A Practical Guide to Support Vector Classification*, <http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf>
15. Support vector machine, http://en.wikipedia.org/wiki/Support_vector_machine

Ant Algorithm Applied in the Minimal Cost Maximum Flow Problem*

Min Xie, Lixin Gao, and Haiwa Guan

Institute of Operations Research and Control Science,
Wenzhou University, Zhejiang, 325000, China
lxgao@wzu.edu.cn

Abstract. The minimal cost maximum flow problem is a classical combinatorial optimization problem. Based on the characteristic of ant algorithm and the minimal cost maximum flow problem, a graph mode is presented to use the ant algorithm to solve the minimal cost maximum flow problem. Simulation results show that the algorithm can efficiently solve minimal cost maximum flow problem in a relatively short time.

Keywords: ant algorithm; minimal cost maximum flow problem; directed network.

1 Introduction

The minimal cost maximum flow problem is an important part of network flows, which is the classical combinatorial optimization problem with many applications such as transportation problem, scheduling problem, etc. Recently, it has been applied in some new domains, such as coding network and wireless ad hoc networks. There are several algorithms which can solve minimal cost maximum flow problem, such as, Cycle-canceling algorithm, Successive shortest path algorithm, Primal-dual algorithm[1]. With the rapid development in the network services, the minimal cost maximum flow problem has recently become a hot spot.

Ant algorithm is first proposed by Dorigo M etc in 1990s, which utilize the similarity of food-seeking behavior of real ants and traveling salesman problem (TSP) and imitate the process of food-seeking behavior of real ants to solve TSP[2]. Ant algorithm has been applied successfully in many applications such as traveling salesman problem[3,4], quadratic assignment problem[5], job-shop scheduling problem[6] and the optimal path planning problem[7]. Actually, ant colony algorithm has been a major concern issue, we attempt to use ant colony algorithm to solve minimum cost maximum flow problem in this paper.

The minimal cost maximum flow problem is a linear programming problem which has close relation with graph theory. The minimal cost maximum flow problem has a structure which is fit to use the ant colony algorithm. In this paper, we attempt to use

* This work was supported by National Nature Science Foundation of China under Grant 60674071.

ant colony algorithm to solve minimum cost maximum flow problem. The remaining part of the paper is organized as follows. In section 2, we introduce the principle of ant colony algorithm. In section 3, we describe minimum cost maximum flow problem and transform the model of minimum cost maximum flow problem based on the characteristic of ant colony algorithm, then use ant colony algorithm to solve it. The computational experiments and results are given in section 4. Finally, we give the summary and the forecast.

2 Ant Algorithm[8-9]

Ant algorithm imitates the behavior of a colony of ants to solve problems. For example, it has been observed that a colony of ants is able to find the shortest path to a food source by marking their trails with a chemical substance called pheromone. As an ant moves and searches for food, it lays down pheromone along its path. As it decides where to move, it looks for pheromone trails and prefers to follow trails with higher levels of pheromone. The ant will lay a higher concentration of pheromone over its path if it takes the shorter path.

The operation of ant system can be illustrated by the classical traveling salesman problem. A traveling salesman problem is seeking for a round route covering all cities with minimal total distance. More formally, TSP can be represented by a complete weighted directed graph $G = (V, E)$ with n nodes, $V = \{1, 2, \dots, n\}$ being the set of nodes, $E = \{(i, j)\}$ being the set of arcs. Suppose there are n cities and m ants. The probability that city j is selected to be visited immediately after city i can be written in a formula as follows:

$$p_{ij}^k = \begin{cases} \frac{[\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{s \in allowed_k} [\tau_{is}]^\alpha \cdot [\eta_{is}]^\beta} & \text{if } j \in allowed_k \\ 0 & \text{else} \end{cases} \quad (1)$$

Where $allowed_k = \{0, 1, \dots, n-1\} - tabu_k$ is the set of cities that have not been visited yet, $tabu_k$ is the set of cities that have been visited, τ_{ij} is the intensity of pheromone trail between cities i and j . $\eta_{ij} = 1/d_{ij}$ is the visibility of city j from city i , d_{ij} is the distance between cities i and j , α is the parameter to regulate the influence of τ_{ij} and β is the parameter to regulate the influence of η_{ij} .

This selection process is repeated until all ants have completed a tour. For each ant the length generated is calculated and the best tour found so far is updated. Then the trail levels are updated as follows: on a tour each ant leaves pheromone quantity given by Q/L_k , where Q is a constant and L_k is the length of its tour. Therefore there is more pheromone left per unit length on shorter tours. By analogy to nature, part of the pheromone evaporates, i.e. the existing pheromone trails are reduced by a factor $(1 - \rho)$ before new pheromone is laid. This is done to avoid early convergence and is

regulated by a parameter ρ . The updating of the trail level τ_{ij} can be written in a formula as follows:

$$\tau_{ij}(t+n) = \rho \times \tau_{ij}(t) + \Delta \tau_{ij} \quad \rho \in (0,1) \tag{2}$$

$$\Delta \tau_{ij}^k = \begin{cases} \frac{Q}{L_k}, & \text{if ant } k \text{ travels on edge } (i, j) \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where t is the iteration counter, $\rho \in [0,1]$ is the parameter to regulate the reduction of τ_{ij} , $\Delta \tau_{ij} = \sum_{k=1}^m \Delta \tau_{ij}^k$, $\Delta \tau_{ij}^k$ is the increase of trail level on edge (i, j) caused by ant k , $\Delta \tau_{ij}$ is the total increase of trail level on edge (i, j) , L_k is the tour length of ant k , m being the number of ants, $\tau_{ij}(t)$, $\Delta \tau_{ij}(t)$, $p_{ij}^k(t)$ can be expressed in different forms, decided according to specific issue, $\tau_{ij}^k(0) = C$ (constant), $\Delta \tau_{ij}^k = 0$ ($i, j = 0, 1, \dots, n-1$).

3 Description of Problem and the Application of Ant Algorithm

3.1 Description of Problem

The minimal cost maximum flow problem is based on directed network. We consider a capacitated network $D = (V, A)$ with a nonnegative capacity c_{ij} and a nonnegative cost b_{ij} associated with every arc $(v_i, v_j) \in A$. To define the minimal cost maximum flow problem, we distinguish two special nodes in the network D , a source node v_s and a sink node v_t . Other nodes are called the intermediate nodes. Generally, this kind of network can be written as $D = (V, A, C, B)$. For each arc (v_i, v_j) , v_i is called a predecessor of v_j , and v_j is called a successor of v_i . We wish to find the minimal cost maximum flow from the source node v_s to the sink node v_t that satisfies the arc capacities and mass balance constraints at all nodes. The minimal cost maximum flow problem can be stated as follows:

$$\text{Minimize } b(f) = \sum_{(v_i, v_j) \in A} b_{ij} f_{ij}$$

Subject to

$$\sum_{(v_i, v_j) \in A} f_{ij} - \sum_{(v_k, v_i) \in A} f_{ki} = \begin{cases} v(f) & (i = s) \\ 0 & (i \neq s, t) \\ -v(f) & (i = t) \end{cases} \tag{4}$$

$$0 \leq f_{ij} \leq c_{ij} \text{ for each } (v_i, v_j) \in A \tag{5}$$

We refer to a vector $f = \{f_{ij}\}$ satisfies (4) and (5) as a flow and the corresponding value of the scalar variable $v(f)$ as the value of the flow.

In order to facilitate and simplify the question, we only consider the situation which capacity and flow are integer. For the case the capacity and the flow are non-integer, we can choose a unit flow, and select a function g_{ij} for each arc (v_i, v_j) which causes it rapid convergence in the suboptimal solution.

3.2 Ant Algorithm Applied in Minimum Cost Maximum Flow Problem

We transform the minimum cost maximum flow problem as follows:

1. Store the nodes which connect with source node into the table S ;
2. For each intermediate node, we select one arc which connects to intermediate node but not to source node, then store the arcs into the table P (to satisfy the constraints better, we select the arcs whose capacity are large enough)
3. Connect the arcs according to the arc subscript order(traverse the nodes according to order from infancy to maturity, then to each node traverse the arcs which connect to them according to order from infancy to maturity),then renumber the nodes with u_1, \dots, u_t according to connection successively order, where the number t is one more than the quantity of table $A - P$;
4. For each arc which are renumbered, produce $c_{ij} + 1$ virtual arcs according to capacity c_{ij} , the capacity of virtual arcs starts from 0 then adds 1 in turn. The flow f_{ij} of arc (v_i, v_j) is the integer of $[0, c_{ij}]$, which are chosen according to following probability choice formula:

$$P_{ijk} = \frac{[\tau_{ijk}]^\alpha}{\sum_{s \in [0, c_{ij}]} [\tau_{ijs}]^\alpha} \quad k \in [0, c_{ij}] \tag{6}$$

where p_{ijk} is the probability of flow of arc (v_i, v_j) equals to k , τ_{ijk} being the intensity of pheromone trail between nodes i and j equals to k .

5. Calculates the flow of table P :

If arc (v_i, v_j) is the output arc, then

$$f_{ij} = \sum_{(v_k, v_i) \in A} f_{ki} - \sum_{\substack{(v_i, v_l) \in A \\ l \neq j}} f_{il} \tag{7}$$

If arc (v_i, v_j) is the input arc, then

$$f_{ij} = \sum_{(v_j, v_k) \in A} f_{jk} - \sum_{\substack{(v_l, v_j) \in A \\ l \neq i}} f_{lj} \tag{8}$$

Minimum cost maximum flow problem is to find a maximum flow f whose cost function $b(f) = \sum_{(v_i, v_j) \in A} b_{ij} f_{ij}$ reach the minimum. Capacities and balance constraints

can be transformed as each arc (v_i, v_j) in table P which satisfy $0 \leq f_{ij} \leq c_{ij}$, then minimum cost maximum flow problem can be transformed as following model:

$$F = \min \sum_{(v_i, v_j) \in A} b_{ij} f_{ij} \tag{9}$$

$$s.t. \quad 0 \leq f_{ij} \leq c_{ij} \quad (v_i, v_j) \in P \tag{10}$$

Now, we illustrate the step 4 in detail, if the capacity of arc (v_i, v_j) equals to 4, then generate 5 virtual arcs, the flow of arc are 0,1,2,3,4 independently.

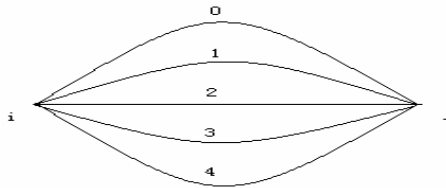


Fig. 1. Virtual path

To solve the minimum cost maximum flow problem, we can imitate real ant’s behavior, set m ants in the source node, each ant use probability choice formula (6) to choose the quantity of flow in table S , until traverse all arcs in table S , then calculate the output flow of source node $v(f) = \sum_{(v_s, v_j) \in A} f_{sj}$, if its value is smaller than the value in the previous iteration (the first iterative compares to initial value), then redistribute the arc flow, until its value is bigger than the value in the previous iterative; then use probability choice formula (6) again to choose arc flow until traverse the table $A - S - P$, then calculate arc flow in table P according to formula (7) or (8). If arc flow of table P satisfy formula (10) and the value of objective function is better than the value in previous iterative (the first iterative compares to initial value) then update the trail level as follow formula

$$\tau_{ijk}(t + n) = \rho \times \tau_{ijk}(t) + \Delta \tau_{ijk} \quad \rho \in (0,1) \tag{11}$$

$$\Delta \tau_{ijk} = \begin{cases} \frac{Q}{F}, & \text{the flow of arc } (V_i, V_j) \text{ equals to } k \\ 0, & \text{others} \end{cases} \tag{12}$$

where F is the value of the objective function

The iteration is repeated until some termination conditions are met, such as a maximum number of iterations has been performed or not get better solution continual h times

If the capacity c_{ij} and arc flow f_{ij} are non-integer, then make the corresponding revision to the step 4 in 3.2, generate $\lfloor c_{ij}/\Delta \rfloor + 1$ virtual arcs according to capacity

c_{ij} , the capacity of virtual arcs starts from 0 then add Δ in turn. The smaller the Δ is, the more precise the solution is.

Algorithm 1

- Step 1. Transform minimum cost maximum flow problem according to 3.2
- Step 2. (parameter initial) Set $nc = 0$ initial maximum iteration NC , initial flow F_0 and initial cost B_0 , set ant number m , set m ants in the source node, $\tau_{ij}(0) = C, \Delta\tau_{ij}(0) = 0$
- Step 3. Choose arc flow in table S according to formula (10) until traversing all arcs in table S
- Step 4. Calculate the output flow $v(f) = \sum_{(v_s, v_j) \in A} f_{sj}$. If $v(f) < F_0$, then go to step 3;
- Step 5. Choose arc flow in table $A-S-P$ according to formula (6) until traversing all arcs in table $A-S-P$, then calculate arc flow in table P according to formula (7) or (8)
- Step 6. Calculate arc flow in table P , if $0 \leq f_{ij} \leq c_{ij}$, then calculate the value of objective function according to formula (9), if its value is better than the value of previous iteration, record the current best solution, $F_0 = v(f), B_0 = F$, else go to step 3, if it is not met the constraint $0 \leq f_{ij} \leq c_{ij}$ continual h times, end the iteration and output the result
- Step 7. Update the best arcs according to global updating formula (11)
- Step 8. Set $\Delta\tau_{ij} \leftarrow 0, nc \leftarrow nc + 1$ to all arcs
- Step 9. If nc is smaller than maximum iteration, go to step 3, else end the iteration and output the result.

4 Numerical Examples

In order to confirm the validity of ant colony algorithm in the minimum cost maximal flow problem, we select the examples of reference [10] and reference [1] to show as figure 2 and Figure 3. The simulation carries on PC machine with the MATLAB. Compare the result which is obtained in this paper with the result which is obtained from [10] and [1]. The selected parameters of examples are shown in Table 1.

Table 1. Parameter setting

α	ρ	C	Q	m	NC	h	F_0	B_0
1	0.2	1	50	20	1000	30	2	1000



Fig. 2. Directed network

Example 1. Consider the minimum cost maximum flow problem shown in figure 2, where vertex v_1 is a source node, vertex v_5 is a sink node, the number of arc is (b_{ij}, c_{ij}) , where b_{ij} shows the cost of unit flow of arc (v_i, v_j) , c_{ij} shows the capacity of arc (v_i, v_j) .

Simulation result is

$$v(f) = 11, F = 55 \quad f_{12} = 3, f_{13} = 8, f_{24} = 0, f_{25} = 7, f_{32} = 4, f_{34} = 4, f_{45} = 4.$$

We can find the minimum cost is 55 and the maximum flow is 11. The simulation result is the same as labeling algorithm in reference [10].

Example 2. Consider the minimum cost maximum flow problem shown in figure 3, where vertex v_1 is a source node, vertex v_7 is a sink node, the number of arc is (b_{ij}, c_{ij}) , where b_{ij} shows the cost of unit flow of arc (v_i, v_j) , c_{ij} shows the capacity of arc (v_i, v_j) .

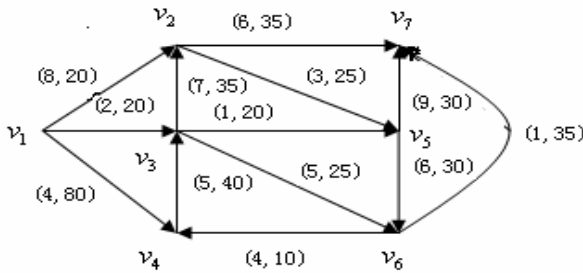


Fig. 3. Directed network

Simulation result is

$$v(f) = 80, F = 1347 \quad f_{12} = 20, f_{13} = 20, f_{14} = 40, f_{25} = 15, f_{27} = 34, f_{32} = 29, f_{35} = 9, f_{36} = 22, f_{43} = 40, f_{56} = 11, f_{57} = 13, f_{64} = 0, f_{67} = 33.$$

We can obtain the minimum cost is 1347 and the maximum flow is 80. The simulation result is the same as the result getting by dual algorithm in reference [1].

The examples 1 and 2 show that the ant colony algorithm is feasible to solve the minimum cost and maximum flow problem.

5 Conclusion

For the minimal cost maximum flow problem, we analyze the characteristic of the minimal cost maximum flow problem, then transform the minimal cost maximum flow problem correspondingly, and use ant colony algorithm to solve the minimal cost maximal flow problem. The simulation results show that the ant colony algorithm can solve the minimum cost maximal flow problem efficiently. In order to facilitate and simplify the question, the scale of simulation selected in this paper is small. This algorithm is also effective to solve the more complex problem. To large capacity situation, we can increase the value of Δ to reduce the complexity of algorithm, but the precision of algorithm will be brought down. As the situation of non-integer and how to choose the unit flow to balance the complexity of algorithm and precision of algorithm will be the content in our further study.

References

1. Ravindra, K., Ahuja, T., James, B.: *Network Flows: Theory, Algorithms, and Applications*. China machine press, Beijing (2005)
2. Dorigo, M.: *Optimization, Learning and Natural Algorithms*, Milano Italy Dipartimento di Elettronica, Politecnico di Milano, Italy (1992)
3. Dorigo, M., Gambardella, L.: Ant Colony System: A Cooperative Learning Approach to the Traveling salesman Problem. *IEEE Transactions on Evolutionary Computation* 1(1), 53–66 (1997)
4. Gambardella, L., Dorigo, M.: Ant-Q: A Reinforcement Learning Approach to the Traveling Salesman Problem. In: *Proceedings of the 12th International Conference on Machine Learning*, pp. 252–260. Morgan Kaufmann, Tahoe City (1995)
5. Maniezzo, V., Colomi, A., Dorigo, M.: The Ant System Applied to the Quadratic Assignment Problem, IRIDIA/94-28. Belgium, Universite de Bruxelles (1994)
6. Colomi, A., Dorigo, M., Maniezzo, V., et al.: Ant System for Job-shop Scheduling. *Belgian Journal of Operations Research, Statistics and Computer Science* 34, 39–53 (1994)
7. Xie, M., Gao, L.: Ant Algorithm Applied in Optimal Path Planning. *Computer Engineering and Application* (2008)
8. Li, S., Chen, Y., Li, Y.: *Ant Colony Algorithms with Applications*. Harbin Industry University Press, Harbin (2004)
9. Duan, H.: *Ant Colony Algorithm: Theory and Applications*. Science Press, Beijing (2005)
10. Ning, X.: A Mixed Labelling Algorithm for Solving Minimum Cost Flow Problem in the Network. *Systems Engineering-theory & Practice* 5(3), 1 (1990)

Computer Simulation and Word-Updating Dynamical Systems (WDS) on Digraphs^{*}

Deqiang Chen¹, Jie Zheng², and Xiaoqian Wu²

¹ College of Information Technology and Science, Nankai University
No.94, Weijin Road, Tianjin 300071, P.R. China
dechen@eyou.com

² Department of Applied Mathematics, Donghua University
No.2999, North Renmin Road, Shanghai 201620, P.R. China
jzheng@dhu.edu.cn, xqwu@dhu.edu.cn

Abstract. Discrete dynamical systems based on dependency digraphs play an important role in the mathematical theory of computer simulation. In this paper, we are concerned with word-updating dynamical systems (WDS) on digraphs, which is a generalization of sequential dynamical systems (SDS) on graphs defined by Barrett et al. By defining an equivalence relation, we obtain the number of different WDS for the given dependency digraph and local functions. It is shown that WDS with the NOR function are closely related to combinatorial properties of the dependency digraphs.

Keywords: Word-updating Dynamical System (WDS); S -pseud Independent Set; Functional Digraph.

1 Introduction

Computer simulation is extensively used for business and science applications. It has become an important tool in the study of complex natural and human-made systems, from the biochemical network underlying cell metabolism to road traffic systems in our cities.

It seems difficult to give a general definition for a computer simulation. However, many examples show the generic structure of computer simulations. One typically finds that in a computer simulation there is a set (finite or infinite) of agents or entities with certain properties or states at a given time. The entities are, roughly, the most refined granularity, or equivalently, the lowest level of aggregation, of the simulated system decomposition. For example, an entity could be a particle in a simulation of a fluid, a vehicle in a transportation simulation, or a trader in a simulation of a market. Any general theory must be insensitive to whether the entities are explicitly or implicitly encoded, and to any other specific representational details related to the entities themselves. The entities can retrieve information from other entities, usually only from those in their own

^{*} Supported by the Chenguang Project in Shanghai City (2008CG40).

vicinity. The entities then update their states based on the states of themselves and the information they retrieve. There will be some kind of scheduling that takes care of the update order.

During the last several years, an effort to establish a rigorous mathematical foundation for computer simulation has been under way. In [1,2,3,4], Barrett et al introduced the concept of sequential dynamical systems (SDS) on graphs, which is motivated by the generic structure of computer simulations. These systems consist of: (a) a graph with vertices $1, 2, \dots, n$, where each vertex has associated with a binary state, (b) a vertex labeled set of functions $F_i (1 \leq i \leq n)$ and (c) a permutation π of the vertices. Every function F_i updates the state of vertex i as a function of the states i and its neighbors and leaves all other states invariant. By composing these functions F_i in the order given by π , the SDS are obtained.

In the SDS, the updating schedule is a permutation of the vertices in the graph, which means, each vertex changes its state once and only once during a global updating. Obviously, it is impracticable. Moreover, the SDS are defined on graphs so that in a computer simulation, the vertices get information from the ones in their own vicinity. But in practice, the process of information exchange may not be bidirectional. That is, a vertex a can get information from a vertex b , but the vertex b may not get information from the vertex a .

These naturally suggest proposing a more generating class of discrete dynamical systems on digraphs. In the systems, the updating schedules are not limited to permutations of the vertices.

We define the word-updating dynamical systems (WDS) on digraphs in Section 2. Some important concepts relative to WDS such as fixed points, periodic points, functional digraph and width are also given. In Section 3, we focus on the following problem. Given the dependency digraph and the local functions, how many different WDS can we get by changing the word? Section 4 is on the properties of *NOR*-WDS, especially the different properties from that of *NOR*-SDS which are extensively studied by Reidys [9]. Any definition not given in the paper, please refer to [6,5,7].

2 WDS on Digraphs

Let $D = (V, A)$ be a digraph with vertex set $V = [n] = \{1, 2, \dots, n\}$ and arc set A , which is called the *dependency digraph*. For each vertex i , $1 \leq i \leq n$, there is a state x_i in some domain \mathbb{D} . Define

$$N_D(i) = \{j \in V | (i, j) \in A\}, \quad d_i = |N_D(i)|, \quad \overline{N_D(i)} = \{i\} \cup N_D(i).$$

Arranging the elements in $\overline{N_D(i)}$ with an increasing order, we get $\overline{N_D(i)}_{<} = (j_1^i, j_2^i, \dots, i, \dots, j_{d_i}^i)$.

There is a local function $f_{i,D}$ over \mathbb{D} associated with each vertex i . It updates the state of the vertex i based on the state of i itself and the states of its out neighbors:

$$f_{i,D} : \mathbb{D}^{d_i+1} \mapsto \mathbb{D}, \quad f_{i,D}(x_{j_1^i}, x_{j_2^i}, \dots, x_i, \dots, x_{j_{d_i}^i}) = y_i. \quad (2.1)$$

Let $F_{i,D}$ be the update function on the global state vectors by applying the local function $f_{i,D}$ to update the state of the vertex i , while keeping other states unchanged, i.e.,

$$F_{i,D} : \mathbb{D}^n \mapsto \mathbb{D}^n, \\ F_{i,D}(x_1, \dots, x_{i-1}, x_i, x_{i+1}, \dots, x_n) = (x_1, \dots, x_{i-1}, y_i, x_{i+1}, \dots, x_n), \tag{2.2}$$

where y_i is the local function (2.1).

Let $\xi = \xi_1 \xi_2 \dots \xi_m$ be a word on $[n]$. That is, $\xi_i \in [n]$ for each $1 \leq i \leq m$. Composing the functions $F_{i,D}$ ($i = 1, 2, \dots, n$) according to a given word $\xi = \xi_1 \xi_2 \dots \xi_m$ on $[n]$, we get a global update function from \mathbb{D}^n to \mathbb{D}^n .

Definition 2.1. Let $D = (V, A)$ be a digraph on $V = [n]$ and $F = \{f_{i,D} \mid 1 \leq i \leq n\}$ be the set of local functions. Then for a word $\xi = \xi_1 \xi_2 \dots \xi_m$ on $[n]$, the mapping

$$[F, D, \xi] = F_{\xi_1,D} F_{\xi_2,D} \dots F_{\xi_m,D} : \mathbb{D}^n \mapsto \mathbb{D}^n \tag{2.3}$$

is called a word-updating dynamical system (WDS) on digraph D .

For (2.3), we assume that the function $F_{\xi_1,D}$ is applied first, $F_{\xi_2,D}$ is applied next, and so on.

If $m = n$, all ξ_i are different and the dependency digraph $D = (V, A)$ satisfies the condition that for any two vertices i, j , either $(i, j) \in A$, $(j, i) \in A$ or $(i, j) \notin A$, $(j, i) \notin A$, then WDS $[F, D, \xi]$ is in fact a SDS defined by Barrett et al (Definition 2 in [1]).

Definition 2.2. For a WDS $[F, D, \xi]$, define a digraph $\Gamma[F, D, \xi]$ as follows. The vertex set of $\Gamma[F, D, \xi]$ is \mathbb{D}^n . There is an arc from X to Y in $\Gamma[F, D, \xi]$ if and only if $[F, D, \xi](X) = Y$. Call $\Gamma[F, D, \xi]$ the functional digraph of WDS $[F, D, \xi]$.

Fixed points and periodic points are two classes of special state vectors in a WDS.

Definition 2.3. For a WDS $[F, D, \xi]$ and a state vector $X \in \mathbb{D}^n$, if $[F, D, \xi](X) = X$, then we call X a fixed point. Denote by $FIX[F, D, \xi]$ the set of fixed points of WDS $[F, D, \xi]$.

Definition 2.4. For a WDS $[F, D, \xi]$ and a state vector $X \in \mathbb{D}^n$, if there exists an integer $m > 1$ such that $[F, D, \xi]^m(X) = X$, then we call X a periodic point. Denote by $PER[F, D, \xi]$ the set of periodic points of WDS $[F, D, \xi]$.

We next define the width of a WDS in terms of its fixed points and periodic points.

Definition 2.5. Let $[F, D, \xi]$ be a WDS on the dependency digraph D . For a state vector $X \in \mathbb{D}^n$, let $h(X)$ be the minimum nonnegative integer such that $g^{h(X)}(X)$ is a fixed point or a periodic point, where $g^0(X)$ is defined to be X . The number $\max\{h(X) \mid X \in \mathbb{D}^n\}$ is called the width of the WDS $[F, D, \xi]$.

3 Counting Different WDS

Definition 3.1. Assume that $D = (V, A)$ is a dependency digraph with vertex set $V = [n]$ and arc set A . $\xi = \xi_1\xi_2 \cdots \xi_m$ is a word on $[n]$. Let \sqsupset_D be such an operation on ξ that changes the positions of two consecutive elements ξ_i, ξ_{i+1} , where $(\xi_i, \xi_{i+1}) \notin A$ and $(\xi_{i+1}, \xi_i) \notin A$.

Definition 3.2. Assume ξ is a word on $[n]$ of a finite length and the number i appears a_i times, $1 \leq i \leq n$. We call the vector (a_1, a_2, \dots, a_n) the type of ξ and denote it by $Type[\xi]$. Conversely, denote by $S[(a_1, a_2, \dots, a_n)]$ the set of words that have type (a_1, a_2, \dots, a_n) .

Example 3.3. Assume $\xi = 44235661234$ and $\zeta = 321532611$. Then

$$Type[\xi] = (1, 2, 2, 3, 1, 2), \quad Type[\zeta] = (3, 2, 2, 0, 1, 1).$$

If $(1, 2, 0, 0, 3)$ is a type, then

$$S[(1, 2, 0, 0, 3)] = \{122555, 155522, 555122, 555221, 212555, 521255, \dots\}.$$

Definition 3.4. Assume D is a digraph on $[n]$. ξ and ζ are two words on $[n]$. If ζ can be obtained from ξ by consecutively applying the operation \sqsupset_D , then we say ξ and ζ have relation \succ_D and denoted by $\xi \succ_D \zeta$. Define $\xi \asymp_D \xi$ for any word ξ .

The relation \asymp_D is obviously an equivalence relation. Therefore, for a set C of words on $[n]$, we can partition C into equivalence classes according to the relation \asymp_D . Denote the set of these equivalence classes by C/\asymp_D . Obviously, $\xi \succ_D \zeta$ implies $Type[\xi] = Type[\zeta]$.

Definition 3.5. Let $\xi = \xi_1\xi_2 \cdots \xi_m$ be a word on $[n]$ and $D = (V, A)$ be a digraph with vertex set $V = [n]$ and arc set A . Define a graph $\Lambda[D, \xi]$ as follows. The vertex set of $\Lambda[D, \xi]$ is the multi-set $\{\xi_1, \xi_2, \dots, \xi_m\}$ and there is an edge in $\Lambda[D, \xi]$ between ξ_i and ξ_j if and only if one of the following conditions is satisfied: (1) $(\xi_i, \xi_j) \in A$, (2) $(\xi_j, \xi_i) \in A$, (3) $\xi_i = \xi_j$.

Example 3.6. Let $D = (V, A)$ be the digraph with $V = \{1, 2, 3, 4\}$ and $A = \{(1, 3), (2, 3), (3, 2), (2, 1), (3, 4)\}$. $\xi = 22314114$. Then by Definition 3.5, $\Lambda[D, \xi]$ is shown as Fig. 1.

Lemma 3.7. There is a bijection between $S[Type[\xi]]/\asymp_D$ and the acyclic orientations of the graph $\Lambda[D, \xi]$.

Proof. Firstly, define a mapping f from $S[Type[\xi]]/\asymp_D$ to the set of the acyclic orientations of $\Lambda[D, \xi]$. Let $[\zeta_1\zeta_2 \cdots \zeta_m]_{\asymp_D}$ be an equivalence class in $S[Type[\xi]]/\asymp_D$. Then the multi-sets $\{\xi_1, \xi_2, \dots, \xi_m\}$ and $\{\zeta_1, \zeta_2, \dots, \zeta_m\}$ are completely same. By the structure of the graph $\Lambda[D, \xi]$, we can label the vertices of $\Lambda[D, \xi]$ such that each vertex has a label in $\{\zeta_1, \zeta_2, \dots, \zeta_m\}$ and different vertices receive different labels. Define that there is an arc from one vertex a to another vertex b in $\Lambda[D, \xi]$ if and only if the following two conditions are satisfied:

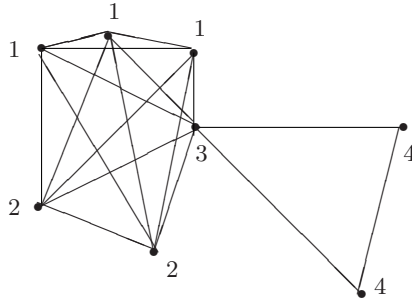


Fig. 1. The graph $\Lambda[D, \xi]$

1. The vertices a and b are adjacent in $\Lambda[D, \xi]$;
2. The label of a is on the left of the label of b in the word $\zeta_1 \zeta_2 \cdots \zeta_m$.

Easy to see that such method to orientate $\Lambda[D, \xi]$ produces no cycles and it is an acyclic orientation. Moreover, if $\rho_1 \rho_2 \cdots \rho_m \succ_D \zeta_1 \zeta_2 \cdots \zeta_m$, then $\rho_1 \rho_2 \cdots \rho_m$ can be obtained by consecutively applying the operation \sqsupseteq_D . By the definitions of \sqsupseteq_D and f , the function f maps $[\rho_1 \rho_2 \cdots \rho_m]_{\succ_D}$ and $[\zeta_1 \zeta_2 \cdots \zeta_m]_{\succ_D}$ to the same acyclic orientation of $\Lambda[D, \xi]$. Then the mapping f is well defined.

Secondly, we prove that f is a surjection. Suppose $\mathfrak{h}[\Lambda[D, \xi]]$ is an acyclic orientation of the graph $\Lambda[D, \xi]$. We can order the labels of the vertices into a word ζ such that the arcs in $\mathfrak{h}[\Lambda[D, \xi]]$ all begin with a left one and point to a right one in ζ . It is obvious that $Type[\zeta] = Type[\xi]$. Then $[\zeta]_{\succ_D}$ is a pre-image of $\mathfrak{h}[\Lambda[D, \xi]]$.

Thirdly, we prove that f is an injection. Assume $[\xi]_{\succ_D}$ and $[\zeta]_{\succ_D}$ are different equivalence classes in $S[Type[\xi]]/\succ_D$. Then ζ can not be obtained from ξ by consecutively applying the operation \sqsupseteq_D . By the definition of \sqsupseteq_D , there must exist $p, q (1 \leq p < q \leq n)$ which satisfy the following two conditions:

1. $(p, q) \in A$ or $(q, p) \in A$;
2. There exist $i, j, i', j' (1 \leq i < j \leq m, 1 \leq j' < i' \leq m)$ such that $\xi_i = p, \xi_j = q, \zeta_{i'} = p$, and $\zeta_{j'} = q$.

Then there is an edge in $\Lambda[D, \xi]$ which receives inverse orientations in $f([\xi]_{\succ_D})$ and $f([\zeta]_{\succ_D})$. Then we have $f([\xi]_{\succ_D}) \neq f([\zeta]_{\succ_D})$.

Therefore, f is a bijection between $S[Type[\xi]]/\succ_D$ and the acyclic orientations of $\Lambda[D, \xi]$. ■

Given the dependency digraph D and local functions $f_{i,D} (1 \leq i \leq n)$, how many different WDS $[F, D, \xi]$ can be obtained by changing the word ξ ? It is not easy to answer since there are infinitely many words on $[n]$ even if n is finite. However, by Lemma 3.7, we have the following result.

Theorem 3.8. *Let $D = (V, A)$ be a dependency digraph on $[n]$ and ξ be a word on $[n]$. Then for any local functions $f_{i,D}, 1 \leq i \leq n$, the number of different WDS $[F, D, \zeta]$ with $Type[\zeta] = Type[\xi]$, is not more than the number of acyclic orientations of $\Lambda[D, \xi]$.*

4 NOR-WDS

Let $Elem[\xi]$ be the set of elements in $[n]$ that appear in the word ξ .

Definition 4.1. *If \mathbb{D} is the Boolean field $\mathbb{B}_2 = \{0, 1\}$ and each local function in $\{f_{i,D} \mid 1 \leq i \leq n\}$ is the Boolean NOR function, that is,*

$$f_{i,D}(x_{j_1}^i, x_{j_2}^i, \dots, x_i, \dots, x_{j_{d_i}}^i) = \overline{x_{j_1}^i \vee x_{j_2}^i \vee \dots \vee x_i \vee \dots \vee x_{j_{d_i}}^i}, \quad (4.1)$$

then we call the WDS in Definition 2.1 a NOR-SDW and denote it by $[NOR, D, \xi]$.

Definition 4.2. *Let $D = (V, A)$ be a digraph and S and T be two subsets of V . If for any two elements i, j in T , $(i, j) \in A$ implies $i \in S$ and $j \in S$, then T is called an S -pseud independent set of the digraph D .*

Lemma 4.3. *For any state vector $X = (x_1, x_2 \dots, x_n) \in \mathbb{B}_2^n$, if*

$$[NOR, D, \xi](x_1, x_2 \dots, x_n) = Y = (y_1, y_2, \dots, y_n),$$

then the set $ID(Y) = \{i \mid y_i = 1, 1 \leq i \leq n\}$ is an $([n] \setminus Elem[\xi])$ -pseud independent set of the digraph $D = (V, A)$.

Proof. Assume that $ID(Y)$ is not an $([n] \setminus Elem[\xi])$ -pseud independent set of D . Then there exist two elements $i, j \in ID(Y)$ such that (i, j) is an arc of D and either $i \in Elem[\xi]$ or $j \in Elem[\xi]$. Without loss of generality, assume $i \in Elem[\xi]$.

- Case 1: $j \notin Elem[\xi]$. Because $j \in ID(Y)$ and there is an arc from i to j in D , the local function $f_{i,D}$ can not map the i -th element to 1. It is a contradiction to the fact that $i \in ID(Y)$.
- Case 2: $j \in Elem[\xi]$. Assume that the last i in the word ξ is on the left of the last j in ξ . Because $i \in ID(Y)$ and $(i, j) \in A$, $f_{j,D}$ can not map the j -th element to 1. Then $j \notin ID(Y)$. A contradiction. ■

Lemma 4.4. *Let $X = (x_1, x_2 \dots, x_n)$ and $Y = (y_1, y_2 \dots, y_n)$ be different state vectors in \mathbb{B}_2^n . If $ID(X) = \{i \mid x_i = 1, 1 \leq i \leq n\}$ and $ID(Y) = \{i \mid y_i = 1, 1 \leq i \leq n\}$ are $([n] \setminus Elem[\xi])$ -pseud independent sets of $D = (V, A)$, then $[NOR, D, \xi](X) \neq [NOR, D, \xi](Y)$.*

Proof. Since $X \neq Y$, we have $ID(X) \neq ID(Y)$. Suppose that $ID(X)$ and $ID(Y)$ are two different $([n] \setminus Elem[\xi])$ -pseud independent sets of D and

$$[NOR, D, \xi](X) = [NOR, D, \xi](Y). \quad (4.2)$$

Let k_0 be the first element in ξ that appears in $ID(X) \cup ID(Y) \setminus (ID(X) \cap ID(Y))$. Without loss of generality, assume that $k_0 \in ID(X)$ and $k_0 \notin ID(Y)$. Then there exists $k_1 >_\xi k_0$ such that $k_1 \in ID(Y)$, $k_1 \notin ID(X)$ and $(k_0, k_1) \in A$. Here

$k_1 >_\xi k_0$ means that there exist k_0 and k_1 such that k_0 is on the left of k_1 in the word ξ . If $k_1 \notin \text{Elem}[\xi]$, then the state of the vertex k_1 will not be updated. Hence $k_1 \in \text{ID}([NOR, D, \xi](Y))$ and $k_1 \notin \text{ID}([NOR, D, \xi](X))$, which is a contradiction to (4.2). If $k_1 \in \text{Elem}[\xi]$, then similarly, for $k_1 \in \text{ID}(X) \cup \text{ID}(Y) \setminus (\text{ID}(X) \cap \text{ID}(Y))$, there exists $k_2 >_\xi k_1$ such that $k_2 \in \text{ID}(X)$, $k_2 \notin \text{ID}(Y)$ and $(k_1, k_2) \in A$. If $k_2 \notin \text{Elem}[\xi]$, then the state of the vertex k_2 will not be updated. Hence $k_2 \notin \text{ID}([NOR, D, \xi](Y))$ and $k_2 \in \text{ID}([NOR, D, \xi](X))$, which is a contradiction to (4.2). By iterating this procedure, we will either get a contradiction or fail at certain step to find $k_i >_\xi k_{i-1}$ such that $k_i \in \text{ID}(X)$, $k_i \notin \text{ID}(Y)$ and $(k_{i-1}, k_i) \in A$, or to find $k_i >_\xi k_{i-1}$ such that $k_i \in \text{ID}(Y)$, $k_i \notin \text{ID}(X)$ and $(k_{i-1}, k_i) \in A$. Hence we get $k_{i-1} \in \text{ID}([NOR, D, \xi](Y))$ and $k_{i-1} \notin \text{ID}([NOR, D, \xi](X))$, or get $k_{i-1} \in \text{ID}([NOR, D, \xi](X))$ and $k_{i-1} \notin \text{ID}([NOR, D, \xi](Y))$. They are both contradictions to the assumption (4.2). ■

The properties of *NOR*-SDS are extensively studied in [9]. From the above two lemmas, we get the following results which have some difference from the properties of *NOR*-SDS due to Reidys [9].

Theorem 4.5. *The width of WDS $[NOR, D, \xi]$ is equal to 1.*

Theorem 4.6. *There is a bijection between $\text{PER}[NOR, D, \xi] \cup \text{FIX}[NOR, D, \xi]$ and the $([n] \setminus \text{Elem}[\xi])$ -pseud independent sets of the digraph D .*

5 Conclusions

To establish a mathematical foundation for computer simulation, we define a new kind of discrete dynamical systems called word-updating dynamical systems (WDS). For the given dependency digraph and local functions, a result is obtained on the number of different WDS. Moreover, we show that the WDS with *NOR* local functions are closely related to combinatorial properties of the dependency digraphs.

References

1. Barrett, C.L., Mortveit, H.S., Reidys, C.M.: Elements of a Theory of Computer Simulation II: Sequential Dynamical Systems. *Appl. Math. Comput.* 107, 121–136 (2002)
2. Barrett, C.L., Mortveit, H.S., Reidys, C.M.: Elements of a Theory of Computer Simulation III: Equivalence of SDS. *Appl. Math. Comput.* 122, 325–340 (2001)
3. Barrett, C.L., Mortveit, H.S., Reidys, C.M.: ETS IV: Sequential Dynamical Systems: Fixed Points, Invertibility and Equivalence. *Appl. Math. Comput.* 134, 153–171 (2003)
4. Barrett, C.L., Reidys, C.M.: Elements of a Theory of Computer Simulation I: Sequential CA over Random Graphs. *Appl. Math. Comput.* 98, 241–259 (1999)
5. Bergeron, F., Labelle, G., Leroux, P.: *Combinatorial Species and Tree-like Structures*. Cambridge University Press, Cambridge (1998)

6. Bollobás, B.: Graph Theory: An Introductory Course. Springer, New York (1979)
7. Bondy, J.A., Murty, U.S.R.: Graph Theory with Its Applications. Elsevier Science Ltd, New York (1976)
8. Mortveit, H.S., Reidys, C.M.: Discrete, Sequential Dynamical Systems. *Discrete Math.* 226, 281–295 (2001)
9. Reidys, C.M.: On Acyclic Orientations and Sequential Dynamical Systems. *Adv. in Appl. Math.* 27, 790–804 (2001)

Cooperative Dynamics in Spatially Structured Populations

Lihui Shang

College of Optics and Electronic Information Engineering,
University of Shanghai for Science and Technology,
Shanghai 200093, China
lhshang2003@163.com

Abstract. The evolution of cooperative behaviors of small-world networking agents in a snowdrift game mode is investigated, where two agents (nodes) are connected with probability depending on their spatial Euclidean lattice distance in the power-law form controlled by an exponent α . Extensive numerical simulations indicate that the game dynamics crucially depends on the spatial topological structure of underlying networks with different values of the exponent α . Especially, in the distance-independent case of $\alpha = 0$, the small-world connectivity pattern contributes to an enhancement of cooperation compared with that in regular lattices, even with a high cost-to-benefit ratio r . However, with the increment of $\alpha > 0$, when $r \geq 0.4$, the spatial distance-dependent small-world (SDSW) structure tends to inhibit the evolution of cooperation in the snowdrift game.

1 Introduction

Understanding the game mechanisms responsible for the emergence and persistence of cooperative behaviors has become one of the central problems in evolutionary biology and socioeconomics [1, 2]. Game theory [3] and the theory of evolutionary games [4, 5] provide a sufficient framework to model individual interactions. Especially, the snowdrift game (SG) [6, 7], an alternative to the prisoner's dilemma game [8] for studying cooperation, has attracted considerable interests.

The SG, also known as the Hawk-Dove game, is originally a two-agent symmetric game, in which each agent can decide to take one of two strategies: cooperate (C) or defect (D). There are four possible combinations: (C, C), (C, D), (D, C), and (D, D) with their payoffs (R, R) , (S, T) , (T, S) , and (P, P) , respectively, satisfying the ordering condition $T > R > S > P$. The best action of one agent depends on his opponent: C is a better strategy than D if his opponent plays D; on the other hand, if his opponent plays C, then D is the best response. Considering now not just two agents but rather a large mixing population of agents, the game finally leads to a mixed evolutionarily stable state [9].

Several years later than the pioneering work of Nowak and May [10], it has been argued that the embedment of spatial interactive structure significantly affects the

cooperative behaviors of the games. Recent studies of the SG played on two-dimensional regular lattices have shown that, the incorporation of spatial dimensional structure may inhibit cooperation [11], and the equilibrium proportion of cooperators is lower than that expected by the game with replicator dynamics [12]. In a more recent work of the SG on a two-dimensional lattice, however, Sysi-Aho *et al.* proposed a different conclusion that, using a simple local decision rule, the cooperation persists through the whole temptation parameter range [13]. The viewpoints that cooperation is sometimes inhibited and sometimes enhanced are also observed on several categories of complex networks ranging from regular lattices to random graphs with small-world networks in between, where the differences are due to different update rules and network contacts [14].

In recent years, spatial Euclidean small-world networks have attracted considerable interests, including but not limited to the navigability [15] and the nature of random walks [16-18]. Moving to the SG, we are interested in uncovering the dependence of cooperative behaviors on the spatial Euclidean structure embedded in underlying small-world networks. In this paper, a spatial distance-dependent small-world (SDSW) network topology is introduced to the SG, in which the length distribution of shortcuts is not uniform, and two nodes are connected according to their spatial lattice distance [19]. We try to explore the cooperative dynamics of the SG whose agents are in the spatial Euclidean situation.

2 SDSW Network Topology

The underlying SDSW network is constructed in a simplified Euclidean space [19]. We begin with a regular two-dimensional lattice with undirected edges as the basic structure. A set of nodes (representing agents in the game) are identified with the set of lattice positions in a $n \times n$ square, $\{(i, j) : i \in \{1, 2, \dots, n\}, j \in \{1, 2, \dots, n\}\}$. We define the lattice distance between two nodes (i, j) and (k, s) as the number of “lattice steps” separating them: $d((i, j), (k, s)) = |k - i| + |s - j|$. We assume every node links to l nearest neighbors (the local contacts), where the constant $l \geq 0$. For a universal constant $q \geq 0$, we add edges with probability p from an arbitrary node u to other q nodes (the long-range contacts). Inspired by the Kleinberg small-world model [15], we select node v as one of the endpoints of q long-range connections of node u with the probability proportional to $[d(u, v)]^{-\alpha}$, where the exponent $\alpha \geq 0$. Therefore, when $\alpha > 0$, two distant nodes are less likely to be connected due to the distance-dependent cost of the edges. Such a graph has almost pqn^2 shortcuts and an average degree $\langle k \rangle \approx l + 2pq$.

In the case of $p = 1$, the above SDSW model reduces to the Kleinberg model, where the number of long-range links for all nodes is the same and determined; while in the case of $l = q$ and $\alpha = 0$, the SDSW model reduces to the Newman-Watts

(NW) small world model [20], where one node’s long-range connections are chosen with uniform probability independently of their positions on the lattice.

The exponent α has a strong impact on both local and global properties of the resulting network. For simplicity, we fix $l = q = 4$. We observe the average path length L and the clustering coefficient CC as functions of the probability p for 50×50 small-world networks with $\alpha = 0$. We can see from Fig. 1 that at first L drops rapidly with the increase of p up to $p \approx 0.2$, at which point $\langle k \rangle \approx 6$, then keeps decreasing slowly with p further increased. However, CC keeps close to zero as p varies, implying that the network with $\alpha = 0$ has almost no local clustering property. For comparison, we plot L and CC as functions of α in Fig. 2 for 50×50 small-world networks with $p = 0.01, 0.2, 1$, respectively, where both L and CC grow linearly with the increment of α , indicating the sensitivity of CC with respect to α , i.e., the network becomes more and more densely clustered, at the same time the average path length also becomes larger.

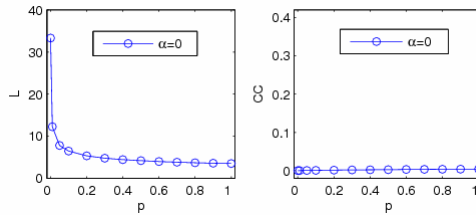


Fig. 1. The average path length L and the clustering coefficient CC as a function of the probability p for small-world networks with $\alpha = 0$

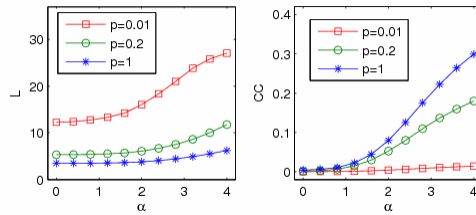


Fig. 2. The average path length L and the clustering coefficient CC as a function of the clustering exponent α for small-world networks with $p = 0.01, 0.2, 1$

3 Spatial Snowdrift Game Model

Following Ref. [11], we rescale the game such that it depends on a single parameter. In the model, we make $T = b > 1$, $R = b - 1/2$, $S = b - 1$ and $P = 0$, such that

the cost-to-benefit ratio of mutual cooperation can be written as $r = 1/(2b - 1)$, where $0 \leq r \leq 1$.

A round of play consists of the encounters of all pairs of individual x and its connected neighbor y , the accumulated payoffs being stored as P_x . The payoffs earned by the agents are not accumulated from round to round. Whenever a site x is updated, a neighbor y is drawn at random among all k_x neighbors; whenever $P_y > P_x$, the chosen neighbor takes over site x with probability given by $(P_y - P_x)/(k_y(T - P))$, where $k_y = \max\{k_x, k_y\}$. In the process of evolution, updating is synchronous where all sites are updated simultaneously through competition with a randomly chosen neighbor.

4 Simulation Results

After extensive numerical simulations, the instance of 50×50 small-world networks is selected as the illustration in the following. We still take $l = q = 4$. The spatial networks are initialized randomly so that each node contains a cooperator or defector with the equal probability. Equilibrium frequencies of cooperation are obtained by averaging over 1000 generations after a transient time of 10000 generations. All data are averaged over 100 groups of network realizations. The network connections, once generated, remain static throughout the evolution of the game.

First, we explore the effect of distance-independent small-world network mechanism with the exponent $\alpha = 0$ on the generic behaviors of the game. Figure 3 shows the frequency of cooperation as a function of r in the SG on small-world networks with different $p = 0, 0.01, 0.1, 0.5, 1$. We can see that the frequency with the small-world topology ($p > 0$) is improved significantly compared with that of the regular lattice ($p = 0$). Clearly, the major contribution to such an improvement arises from the adding of shortcuts in the population structure (taking place with increasing p). Our simulations also suggest that the dependence of the results on p is not monotonic, which is shown clearly in the inset of Fig. 3.

We plot the frequency of cooperation as a function of p for different r in Fig. 4, since the small-world probability is important for the evolution of cooperation. Remarkably, we find that with the independence of r , the frequency keeps increasing with respect to the probability p up to $p \approx 0.2$, where the performance is already greatly enhanced. This indicates that the overall incidence of cooperators is sensitive to the average path length L , which has decreased typically one order of magnitude from its value at $p = 0$ as previously shown in Fig. 1. For $r = 0.3$ and $r = 0.4$, there exists a pronounced frequency peak at $p \approx 0.2$, illustrating an

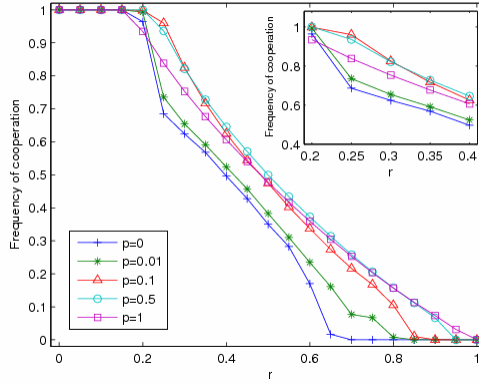


Fig. 3. Frequency of cooperation as a function of the cost-to-benefit ratio r in the snowdrift game on small-world networks with the exponent $\alpha = 0$ and the probability $p = 0, 0.01, 0.1, 0.5, 1$

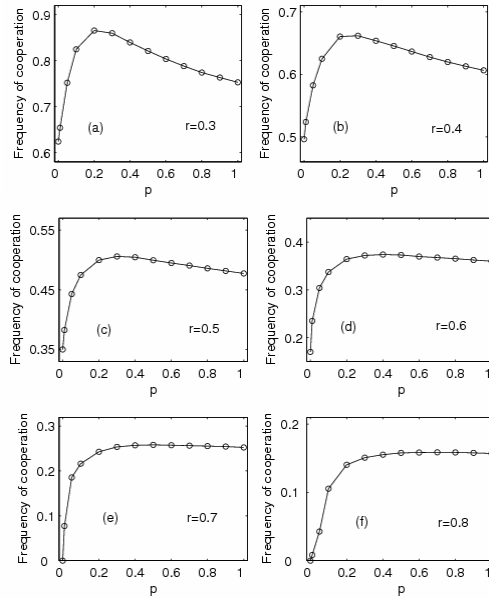


Fig. 4(a)-(f). Frequency of cooperation as a function of the probability p in the snowdrift game on small-world networks with the exponent $\alpha = 0$ and the cost-to-benefit ratio $r = 0.3, 0.4, 0.5, 0.6, 0.7, 0.8$

optimal cooperation occurs at this point. With the increment of r , the peak becomes more and more saturated, and no further qualitative changes take place for $p > 0.2$, owing to the fact that L varies a little at this stage.

Now, we investigate how the exponent α in the SDSW networks influences the evolution of cooperation. As described above, the increment of α marks the onset of a more rapid change of the clustering property and a larger average path length. Only for small r ($r < 0.3$) is the proportion of cooperators not affected by the changes of α , mainly due to the high benefit and low cost. However, when r is larger, the spatial structure with increased α affects the cooperative dynamics greatly. The frequency of cooperation as a function of α for the SG with $p = 0.01, 0.2, 1$ and $r = 0.3, 0.6$, respectively, is shown in Fig. 5. When $r < 0.4$, the benefit is lower but still relative high, which leads to complex behaviors with different p . A typical case is shown in the left panels of Fig.5 corresponding to $r = 0.3$, where as α grows from zero, the equilibrium proportion of cooperators decreases for $p = 1$, increases for $p = 0.2$, and remains almost unchanged for $p = 0.01$. When $r \geq 0.4$, the benefit becomes less, the spatial clustered network makes cooperators be easily invaded by defectors, which favors defectors and tends to inhibit cooperation. A typical example is shown in the right panels of Fig.5 corresponding to $r = 0.6$, where the results for $p = 0.01, 0.2, 1$, respectively, are essentially identical, and the equilibrium proportion of cooperators drops with the increment of α , where the long-distance connections become more and more sparse, and

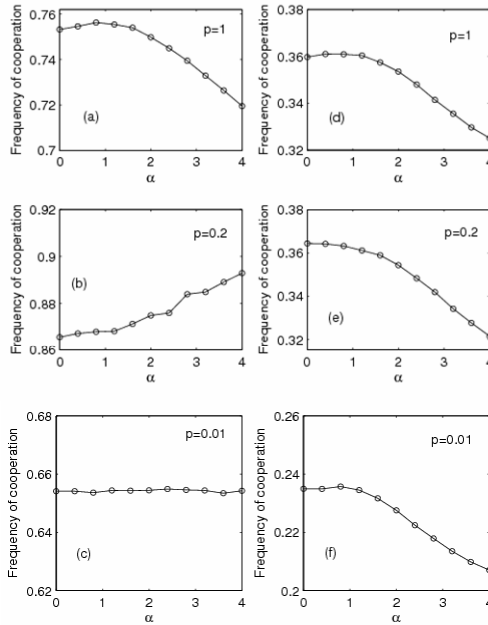


Fig. 5. Frequency of cooperation as a function of the exponent α for the snowdrift game on small-world networks with the probability $p = 0.01, 0.2, 1$ and the cost-to-benefit ratio $r = 0.3, 0.6$. (a)-(c) correspond to $r = 0.3$ and (d)-(f) correspond to $r = 0.6$.

short-distance connections become more and more dense in the network, indicating long-distance connectivity pattern benefits overall cooperation in this high cost-to-benefit spatial situation. Our extensive simulations with other population sizes also support this turning point of cooperation at $r = 0.4$, which is independent of the population size.

5 Conclusion

Cooperative behaviors of the snowdrift game are sensitive to the underlying spatial network structures. In particular, when $\alpha = 0$, the spatial distance-independent small-world mechanism contributes to an enhanced level of cooperation among the populations compared with regular lattices, even with a very high cost-to-benefit ratio. When $\alpha > 0$, and $r \geq 0.4$, the SDSW network structure tends to inhibit the evolution of cooperation in the snowdrift game.

References

1. Von Neumann, J., Morgenstern, O.: *Theory of Games and Economic Behavior*. Princeton University Press, Princeton (1944)
2. Smith, M.J., Szathmáry, E.: *The Major Transitions in Evolution*. W. H. Freeman, Oxford (1995)
3. Duncan Lucc, R., Raiffa, H.: *Games and Decisions*. Dover, New York (1985)
4. Maynard Smith, J.: *Evolution and the Theory of Games*. Cambridge University Press, Cambridge (1982)
5. Gintis, H.: *Game Theory Evolving*. Princeton University, Princeton (2000)
6. Maynard Smith, J., Price, G.: *The Logic of Animal Conflict*. *Nature* 246, 15–18 (1973)
7. Sugden, R.: *The Economics of Rights, Co-operation and Welfare*. Blackwell, Oxford (1986)
8. Axelrod, R.: *The Evolution of Cooperation*. Basic Books, New York (1984)
9. Doebeli, M., Hauert, C.: *Models of Cooperation Based on the Prisoner's Dilemma and the Snowdrift Game*. *Ecology Letters* 8, 748–766 (2005)
10. Nowak, M.A., May, R.: *Evolutionary Games and Spatial Chaos*. *Nature* 359, 826–829 (1992)
11. Hauert, C., Doebeli, M.: *Spatial Structure often Inhibits the Evolution of Cooperation in the Snowdrift Game*. *Nature* 428, 643–646 (2004)
12. Hofbauer, J., Sigmund, K.: *Evolutionary Games and Population Dynamics*. Cambridge University Press, Cambridge (1998)
13. Sysi-Aho, M., et al.: *Spatial Snowdrift Game with Myopic Agents*. *Eur. Phys. J. B.* 44, 129–135 (2005)
14. Tomassini, M., Luthi, L., Giacobini, M.: *Hawks and Doves on Small-world Networks*. *Phys. Rev. E* 73, 016132 (2006)
15. Kleinberg, J.M.: *Navigation in a Small World*. *Nature* 406, 845–847 (2000)
16. Jespersen, S., Blumen, A.: *Small-world Networks: Links with Long-tailed Distribution*. *Phys. Rev. E* 62, 6270–6274 (2000)

17. Sen, P., Chakrabarti, B.K.: Small-world Phenomena and the Statistics of Linear Polymers. *J. Phys. A* 34, 7749–7755 (2001)
18. Sen, P., Banerjee, K., Biswas, T.: Phase Transition in a Network with a Range-dependent Connection Probability. *Phys. Rev. E* 66, 037102 (2002)
19. Boccaletti, S., et al.: Complex Networks: Structure and Dynamics. *Physics Reports* 424, 175–308 (2006)
20. Newman, M.E.J., Watts, D.J.: Scaling and Percolation in the Small-world Network Model. *Phys. Rev. E* 60, 7332–7342 (1999)

Emergent Motion Characteristics of a Modular Robot through Genetic Algorithm

Sunil Pranit Lal, Koji Yamada, and Satoshi Endo

Complex Systems Laboratory, Department of Information Engineering,
Faculty of Engineering, University of the Ryukyus,
1 Senbaru, Nishihara, Okinawa 903-0213, Japan
sunil@eva.ie.u-ryukyu.ac.jp, {koji, endo}@ie.u-ryukyu.ac.jp

Abstract. In this paper we present an approach to transform individual behaviour of homogenous sub-systems to yield desired global behaviour of the overall system. As a test bed we consider the brittle star robot as the system which is composed of homogenous modules (sub-systems). Using genetic algorithm, the rotational motion of the individual modules is translated into rectilinear motion of the robot. We argue that given a set of sub-system level behaviours, it is better to discover intermediate system level infinitesimal behaviours as a stepping stone to developing desired system level global behaviour.

Keywords: Emergence; behaviour; motion control; modular robot; genetic algorithm.

1 Introduction

The notion that the whole is greater than the sum of the parts is very applicable when considering the concept of emergence in complex systems. Emergent structures in nature, such as ant colonies, neural networks and cellular automata provide inspiration for developing equivalent computational models for solving difficult problems [1]. Given a complex system, in our case a modular robot, we are interested in studying how the local interactions of the modules can lead to emergence of locomotion. For this purpose we leverage off genetic algorithm (GA) as a means to transform the individual modular behaviour into desired system level behaviour

Inspired by biological adaptations, genetic algorithm is essentially a search technique used extensively to solve optimization related problems [2]. In the literature many notable contributions have been made in the field of robotics and control using evolutionary approach. Kamimura et al. [3] used neural oscillator as a central pattern generator for controlling a modular robot (M-TRAN II). The optimal parameters for the pattern generator were evolved using GA. Reil and Husbands [4] successfully simulated bipedal straight-line walking using recurrent neural network whose parameters were evolved by GA. Porting genetically evolved neural network controller for a hexapod robot from simulation model to

actual hardware was demonstrated by Gallagher et al. [5]. One of the conclusions reached by them was that the evolved controller performed extremely well in real world in spite of the fact that inertia, noise and delays were not taken into account in the simulation.

The motion of the brittle star-typed robot considered in this paper requires coordinated movement of the modules. In prior research, cellular automata-based control model [6] and neural network control model [7] were developed. The parameters evolved for these control models were highly specific to the given task such as rectilinear locomotion. From this we came to the realization that if the robot is required to traverse any arbitrary path, a new set parameters need to be evolved each time, which is quite a time consuming process. This has lead us to develop a novel approach where by the infinitesimal movements of the robot is evolved from the interactions of the individual modules and then these infinitesimal movements can be logically combined to produce desired global motion characteristics of the robot. The effectiveness of the proposed approach was verified using simulation model of the robot developed using Open Dynamics Engine [8]

The paper is organized as follows: The decomposition of the motion control problem in terms of state transitions is presented in Sect. 2. Following it is a descriptive account of evolution of infinitesimal motion patterns using GA in Sect. 3. Section 4 discusses a logical method of combining the infinitesimal movements to produce desired motion characteristics, and finally Sect. 5 concludes this paper.

2 Motion Control Problem Formulation

Given a system consisting of homogenous sub-systems with a set of behaviour and constraints, we are interested in knowing how the interaction of the sub-systems can lead to emergence of system level behavior. In particular we consider the brittle star robot (Fig. 1(a)) as the system which is composed of homogenous modules (sub-systems). Each module (Fig. 1(b)) incorporates an onboard micro controller (BASIC stamp 2sx), actuator (RC Servo Futaba S5301) and two touch sensors.

Moving the robot requires coordinated movement of the individual modules. Thus the problem is how to transform the individual behaviour of modules (rotational motion) into emergent global behaviour of the robot, such as walking in a straight line.

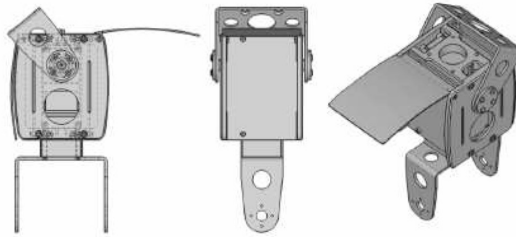
One way to visualize the module is to consider it as a state machine. Since the motion of any given module is basically rotation between an allowable range, the state of j^{th} module in the i^{th} leg at time, t is given by (1).

$$S_{ij}^t \in \{\theta \mid \theta_{min} \leq \theta \leq \theta_{max}\}. \quad (1)$$

This makes the robot a collection of state machines, or to put it differently, a state machine of state machines. The state of the robot with n legs and m modules per leg at time, t is thus given by (2).



(a)



(b)

Fig. 1. (a) The brittle star robot (b) Individual module connected to make up the leg

$$R^t = \{S_{ij}^t \mid 0 \leq i < n, 0 \leq j < m\}. \quad (2)$$

Therefore the problem of motion control of the robot becomes the task of finding optimal sequence (O_T) of state transitions, which would transform the robot producing desired locomotion.

$$O_T = R^0, R^1, R^2, \dots, R^T. \quad (3)$$

It should not take much imagination to realize the sheer magnitude of the search space that needs to be explored to find near optimal solution, and thus the need for evolutionary computational approach.

3 Evolving Motion Characteristics Using Genetic Algorithm

Genetic algorithm or for that matter any other evolutionary algorithm provide solution which is highly suited to a specific problem. For instance, using GA to find optimal sequence of state transitions of the robot to traverse a given path will generate a sequence for that specific path, and as such it will be futile for any arbitrary path.

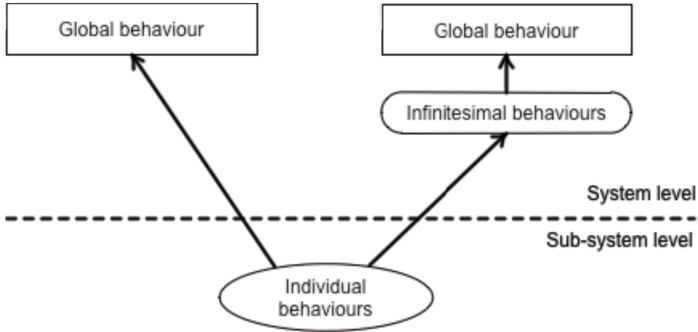


Fig. 2. Approaches in deriving global behaviour from individual behaviour

Differing from conventional wisdom, our approach (Fig. 2) is to evolve infinitesimal system level behaviour from sub-system level behaviour and then logically combine the infinitesimal behaviour to yield desired global behaviour of the system.

Given the dimensions of the robot, length of a single leg being 54cm, it was decided that 2cm would suffice as the infinitesimal straight line distance. While the robot can move this distance in any direction, we decided to limit it to 8 directions corresponding to the cardinal and primary inter-cardinal points on a compass.

3.1 Genetic Encoding

We begin by discretizing the angular range $[-\frac{\pi}{3}, \frac{\pi}{3}]$ of the modules into 16 states which provided fairly granular control of the robot. It is not known a priori the number of state transitions (R^t) required for each of the eight possible infinitesimal behaviour corresponding to robot's movement in eight direction, thus variable length chromosome was used. To enable smooth and continuous motion across any combination of infinitesimal behaviours, the robot is required to start at predetermined initial state and return to that initial state at the end of state transition for any infinitesimal behaviour (Fig. 3). Finally, the state transitions for movement in each direction was evolved separately and independent of each other.

With 30 modules in the robot and 16 states per module, the equivalent binary representation of the state of the robot requires 120 bits. The state search space grows exponentially (2^{120q}) with the number of states (q) encoded in the chromosome.

3.2 Fitness Function

The fitness of each chromosome is evaluated by applying state transitions encoded in the chromosome to the simulated model of the robot. The robot starts

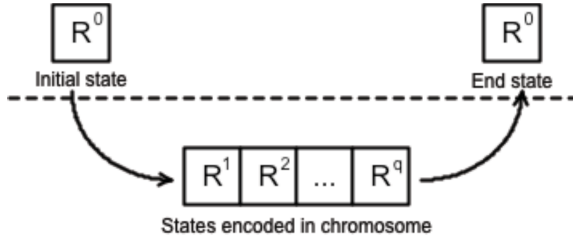


Fig. 3. State transition from predetermined initial state through intermediate states encoded in the chromosome, and back to the initial state

at an initial position and is required to move towards a fixed target position located 2cm away from the initial position in the desired direction.

As the robot moves we would like to minimize the deviation of the robot from the desired path to the target, as well to minimize the euclidean distance between the robot's final position and the target position (Fig. 4). In addition it is highly desirable to minimize the number of state transitions as doing so not only reduces the search space but also improve the robot's performance in terms of speed in reaching the target. Taking these issues into consideration the derived fitness function (F), which needs to be minimized, is a function of deviation from path (D_{path}), deviation from target (D_{target}) and number of states (N) in excess of minimum number of states (N_{min}) that can be encoded in the chromosome.

$$F = D_{path} + D_{target} + N . \quad (4)$$

$$D_{path} = \mathbf{max} |l\sin(\phi - \theta)| . \quad (5)$$

$$D_{target} = \sqrt{(x_T - x_R)^2 + (y_T - y_R)^2} . \quad (6)$$

$$N = N_{actual} - N_{min} . \quad (7)$$

3.3 Genetic Operators

Based on the fitness of the chromosomes in the population, GA operations; namely selection, crossover and mutation are applied to the whole population. The selection process screens the individuals such that the fitter individuals have higher probability of making it through to the next generation. The roulette wheel selection method was used to perform selection. It is worth mentioning that the roulette wheel selection method is customized for maximization problems, thus the fitness function, F was transformed to $F' = -F + C$, where C is positive constant.

The crossover operation essentially emulates mating process by exchanging and combining genes from selected parent chromosomes to produce offsprings,

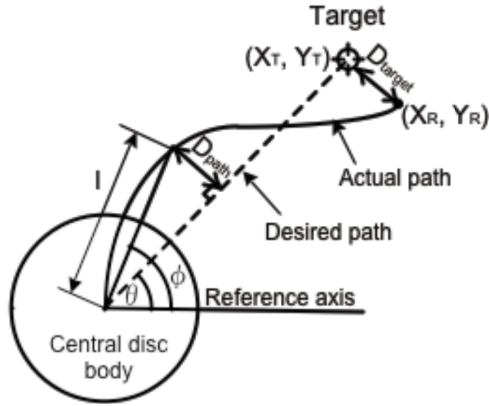


Fig. 4. Derivation of the fitness function

with the hope that they may have better fitness than the parents. The selected pairs of chromosomes are crossed over using one-point crossover at randomly chosen locus with a crossover probability of P_C . Crossover operation is skipped for a chromosome pair if the offspring produced would exceed the maximum chromosome length in terms of maximum number of states encoded (N_{max}).

Mutation is a way to introduce diversity within the population, thus enabling better exploration of the search space. After the selection and crossover operation, a two fold mutation operation was applied to the entire population with a mutation probability, P_M . Given the binary representation and variable length of the chromosome, the mutation operation involved random bit flips, and resizing chromosome between $[N_{min}, N_{max}]$ by deleting states from chromosome at random position, or inserting randomly created states at random positions.

3.4 Simulation Results

The simulation was carried out over numerous trails involving different target positions, and using parameters shown in Table 1. In each trial, initial population of chromosomes of size (POP_SIZE) was randomly generated and GA was executed for a number of generations (MAX_GEN). For each generation, the fitness of all the chromosomes in the population is evaluated after which genetic operators are applied to the population to create the next generation.

The motion characteristics of the best evolved state transitions in terms of amount of deviation from path and target is highlighted in Table 2. The fitness of the best chromosomes approached the optimal value of zero. The small differences in the robot's performance can be attributed to the fact that the orientation of the pentagonal robot varies with respect to the frame of reference for each of the direction of movement. For example, to move in 0° direction, 3 legs face forward and 2 legs face backward, however to move in 180° direction, 2 legs face forward and 3 legs face backward, thus the resulting differences in the motion characteristics.

Table 1. Summary of simulation parameters

Parameter	Value
P_C	0.80
P_M	0.005
N_{min}	1
N_{max}	6
POP_SIZE	55
MAX_GEN	400

It is worth mentioning that at the beginning of GA, the best chromosomes had variable lengths, however towards the end all of them converged to encoding just a single state of the robot. In other words, including the predetermined initial and final states, the total number of state transitions required to effect motion of 2cm in any of the directions turned out to be three.

Table 2. Characteristics of the best evolved state transitions for each of the directions

Direction (degrees)	Fitness	D_{path} (mm)	D_{target} (mm)
0°	1.40	1.03	0.37
45°	1.70	1.30	0.40
90°	1.32	1.02	0.30
135°	1.39	1.23	0.16
180°	2.00	1.46	0.54
225°	3.20	1.98	1.22
270°	1.83	1.68	0.15
315°	2.51	1.61	0.90

4 Discussion

In this section we consider a simple method of logically combining the “infinitesimal” movements in producing desired motion. Basically the goal is for the robot to traverse along a specified path. At any given point in time along the path, the robot has a set eight directions to choose from. The control strategy adopted involves determining the position of the center of the robot in each of the eight directions at the next step. Among these positions, the position which has the least deviation from the path and is in the direction of travel, is chosen as the direction in which the robot will move next (Fig. 5).

The state transitions evolved in the previous section focused on rectilinear movements of 2cm. We are interested in knowing if these infinitesimal movements can be effectively combined to cover greater distances. Given the task of moving in a straight line distance of 20cm, Fig. 6 shows the simulation results of robot using combination of infinitesimal movements compared with a robot using state transitions specifically evolved to traverse the given path. As can be expected, the genetic algorithm yields near optimal state transition for the specific path, thus

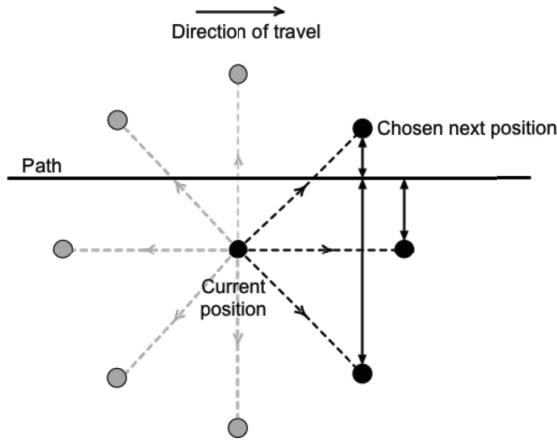


Fig. 5. Control strategy in choosing the next move

the specially evolved robot outperforms the robot with combined infinitesimal movements in terms of total number of state transitions to reach the target. The small errors associated with the individual infinitesimal movement (Table 2) compound to significant error margins when those movements are combined, thus shedding some light on the observed fluttering behaviour of the robot around the desired path as it constantly adjusts its course to stay on the path.

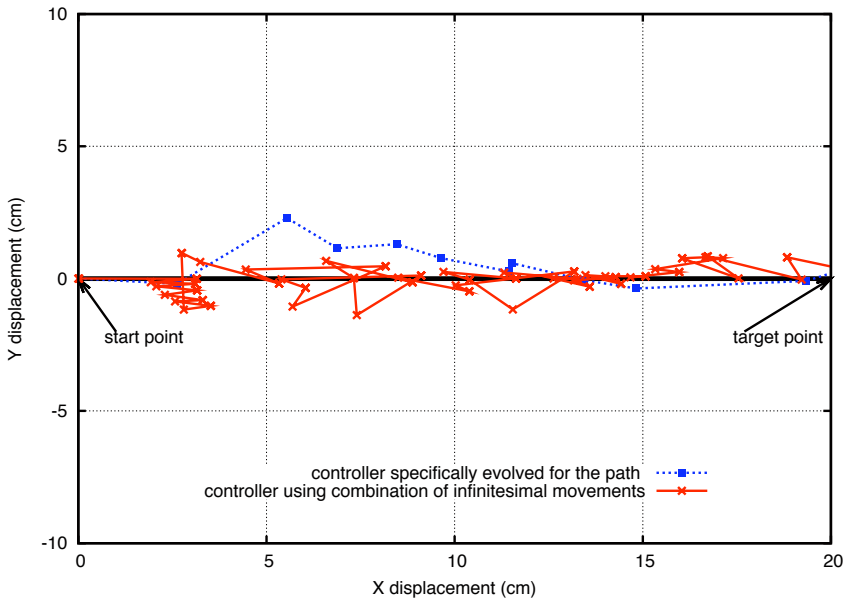


Fig. 6. Displacement of the central point of the robot using different control strategies

While the motion characteristics produced by combining infinitesimal movements is far from optimal, a strong argument in favor of this approach is that it is highly adaptable to different problems. Consider a scenario where the robot is required to traverse an arbitrary path. Using GA to evolve a set of state transitions for that path, aside from consuming generation after generation of computational time, will yield a result which is specifically suited to that path and will be of no value in traversing some other path. On the other hand, once the infinitesimal movements have been evolved they can be combined in a number of ways to adapt to any arbitrary path without requiring any additional training time as shown in Fig. 7.

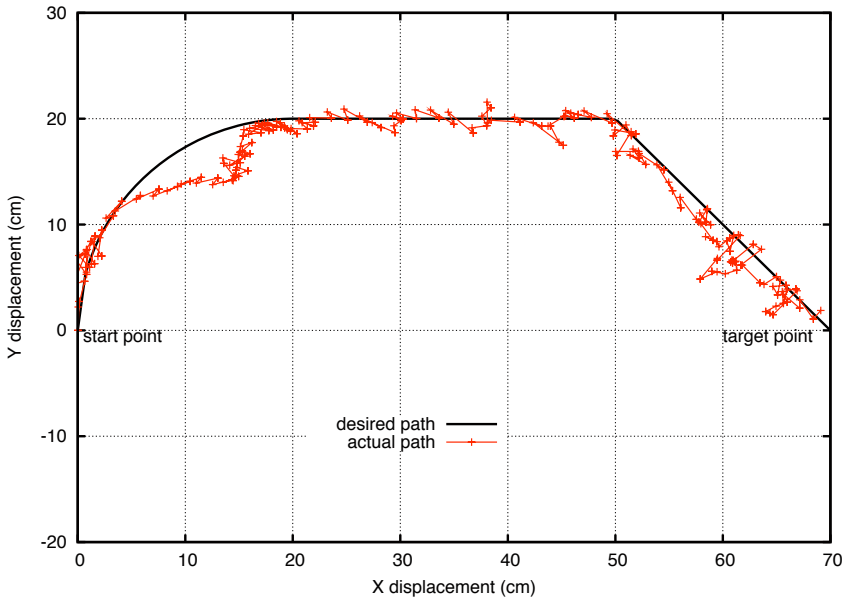


Fig. 7. Displacement of central point of the robot using combination of infinitesimal movements to traverse an arbitrary path

5 Conclusion

In this paper we considered the emergence of global behaviour of a complex system consisting of homogenous sub-systems. Specifically we studied a modular robot as an instance of a complex system in an effort to derive global motion characteristics of the robot from individual modular behaviour. Our proposed approach consisted of evolving infinitesimal rectilinear movements from rotational motion of individuals modules, and then logically combining the infinitesimal movements to produce desired motion characteristics of the robot. This approach, though far from optimal, was highly adaptable to any arbitrary path that the robot was required to traverse.

The infinitesimal behaviour of the robot consisting of small rectilinear movements in eight directions was rather elementary. Additional infinitesimal behaviours such as clockwise and counter clockwise rotation can be added to increase the set of movements available to the robot. In combining the infinitesimal movements, the errors compounded causing performance degradation of the robot. Future work towards implementing the control strategy using controllers that can operate well in the presence of uncertainty such as fuzzy logic controller should prove fruitful.

References

1. Holland, J.H.: *Emergence: From Chaos to Order*. Addison Wesley, Reading (1998)
2. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading (1989)
3. Kamimura, A., Kurokawa, H., Yoshida, E., Tomita, K., Murata, S., Kokaji, S.: Automatic Locomotion Pattern Generation for Modular Robots. In: *IEEE International Conference on Robotics and Automation*, pp. 714–720. IEEE Press, New York (2003)
4. Reil, T., Husbands, P.: Evolution of central pattern generators for bipedal walking in a real-time physics environment. *IEEE Transactions on Evolutionary Computation* 6(2), 159–168 (2002)
5. Gallagher, J.C., Beer, R.D., Espenschied, K.S., Quinn, R.D.: Application of Evolved Locomotion Controllers to a Hexapod Robot. *Robotics and Autonomous Systems* 19, 95–103
6. Lal, S.P., Yamada, K., Endo, S.: Studies on Motion Control of a Modular Robot Using Cellular Automata. In: Sattar, A., Kang, B.-H. (eds.) *AI 2006. LNCS (LNAI)*, vol. 4304, pp. 689–698. Springer, Heidelberg (2006)
7. Lal, S.P., Yamada, K., Endo, S.: Evolving Motion Control for a Modular Robot. In: Ellis, R., Allen, T., Petridis, M. (eds.) *Applications and Innovations in Intelligent Systems XV*, pp. 245–258. Springer, London (2007)
8. Smith, R.: *Open Dynamics Engine User Guide*, <http://www.ode.org/>

Evolutionary Optimization with Dynamic Fidelity Computational Models

Dudy Lim¹, Yew-Soon Ong¹, Yaochu Jin², and Bernhard Sendhoff²

¹ School of Computer Engineering, Nanyang Technological University, Singapore

² Honda Research Institute Europe GmbH, Offenbach/Main, Germany

Abstract. In this paper, we propose an evolutionary framework for model fidelity control that decides, at runtime, the appropriate fidelity level of the computational model, which is deemed to be computationally less expensive, to be used in place of the exact analysis code as the search progresses. Empirical study on an aerodynamic airfoil design problem based on a Memetic Algorithm with Dynamic Fidelity Model (MA-DFM) demonstrates that improved quality solution and efficiency are obtained over existing evolutionary schemes.

1 Introduction

In science and engineering, computational models serve as efficient and convenient alternatives for conducting studies on the original real-world system that are otherwise deemed to be too costly or hazardous to construct. A motivating real world example for us is the aerodynamic aircraft wing analysis and design where various multi-fidelity models ranging from low-fidelity simple-physics models to high-fidelity detailed simulation models have been studied [1,2,3,4,5,6]. In this field, Navier-Stokes and Euler equations are examples of two computational models with different fidelity, where the latter is obtained by removing the viscosity terms from the Navier-Stokes equation. The term “fidelity” here refers to the extent to which a model is capable to mimic the original physical system of interest. A common assumption is that higher fidelity models are generally more accurate at the expense of a higher computational cost. The complexity and level of details of a physical system may exist in many forms, from variable mathematical models [7], variable parametric formulations [4,8,9,10], variable operating conditions [11] to variable residual tolerance levels [12].

In the context of evolutionary optimization, it is possible to categorize existing evolutionary algorithms [13] that employ variable fidelity computational models into 1) non-adaptive, i.e., one or more fixed low fidelity model(s) in tandem with the original expensive computational model, or 2) deterministic adaptive approaches, i.e., pre-defined adaptation rule(s) is(are) used to manage the variable fidelity models employed. In the first category, [9] employs a fixed lower resolution computational model during the evolutionary design of a communication antenna. Subsequently, the obtained solution is then refined by space mapping. On the other hand, instances of deterministic adaptation can be found in [4,8,10]. In [4], the evolutionary search employs computational models of different grid mesh that vary from coarse to fine grids at pre-defined stages of the GA search. In a multi-island GA optimizer [8,10], multiple subpopulation of individuals are equipped with models of variable fidelity levels. Individuals then

migrate across subpopulation which are subsequently evaluated based on the computational model of the respective subpopulation. Last but not least, [7] uses a pre-defined adaptation rule to regulate the proportions of individuals in the population with different fidelity models in the evolutionary search.

In this paper, our objective is to present an investigation on evolutionary framework that employs *dynamic fidelity* computational models for solving problems with computationally expensive objective functions. In particular, we propose a evolutionary framework for model fidelity control that decides, at runtime, the complexity level of the localized computational model to be used in place of the exact analysis code as the search progresses. To the best of our knowledge, the present study represents a first attempt on dynamic adaptation in evolutionary optimization where dynamic localized fidelity computational models have been deployed for solving problems with computationally expensive objective functions.

The remaining of this paper is organized as follows. Section 2 describes a multi-fidelity airfoil computational model used in the present study for aerodynamic design. Subsequently, Section 3 introduces the memetic evolutionary framework with dynamic fidelity computational models for optimizing the computationally expensive aerodynamic design problem. Section 4 summarizes the empirical results on the aerodynamic design problem and analyzes the results with comparison to existing evolutionary optimizers. Finally, Section 5 provides a brief conclusion of this paper.

2 Multi-fidelity Models in the Aerodynamic Airfoil Design Problem

In this section, we provide an example of multi-fidelity computational models, prevalent in many real-world problems, using the aerodynamic airfoil design problem. For illustration purpose, Fig. 1 depicts an aircraft together with an airfoil, which is simply the cross-section of the aircraft wing (see Fig. 1). Particularly, we consider the parametric design optimization of 2D airfoil structure using a subsonic inverse pressure design problem. In the inverse design problem, the aim is to minimize the difference between the surface pressure P of a given airfoil with the desired pressure profile P_d of a baseline shape. If W is the flow variables and S the shape design variables, the inverse pressure design problem can be formulated as a minimization problem of the form:

$$f(W, S) = \frac{1}{2} \int_{wall} (P - P_d)^2 d\sigma \quad (1)$$

For the purpose of this study, the target pressure profile is generated from the NACA 0015 airfoil as the baseline shape. The airfoil geometry used here is characterized using 24-parameter Hicks-Henne representation. Since we only consider compressible non-viscous flow, a finite-volume Euler solver with body-fitted grid and explicit time-stepping is employed for the purpose of this study. The free-stream conditions in this problem are subsonic speed of Mach=0.5, and angle of attack (AOA)=2.0, corresponding to symmetric pressure profiles on the upper and lower walls.

In practice, the evaluation of a single individual design (refer to equation 1) is an iterative process, usually terminated after a certain residual error threshold has been

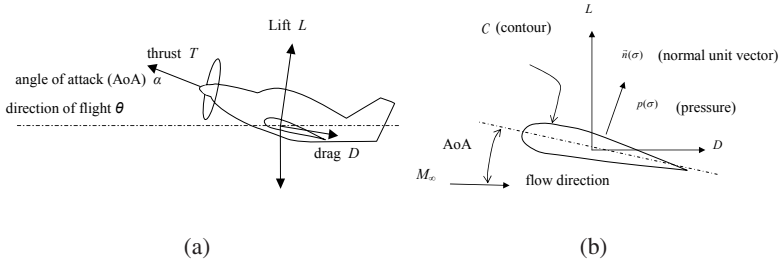


Fig. 1. (a) An aircraft and (b) an airfoil, the cross-section shape of the wing

reached. In the case of limited computational budget, it is also possible to have a maximum number of iterations as the effective termination condition. Hence, variable fidelity models can be obtained from partial results at different levels of residual error tolerance and/or maximum iterations. Generally, lower residual error tolerance and greater number of iterations provide higher fidelity outputs.

The correlation factor (r) and root mean square error ($rmse$) of the models at variable levels of residual error tolerance (i.e., fidelity) to the original high fidelity airfoil analysis model are reported in Figs. 2(a) and 2(b), respectively. In this study, a set of design points, obtained from space-filling Design of Experiment (DOE) sampling, are partitioned into separate local clusters based on their proximity in the search space of the airfoil model. The correlation factor, r_k^j , and root mean square error, $rmse_k^j$ for data cluster j at fidelity level k , are defined as follows:

$$r_k^j = \frac{n \sum_{i=1}^n f_i \hat{f}_i - \sum_{i=1}^n f_i \sum_{i=1}^n \hat{f}_i}{\sqrt{[n \sum_{i=1}^n f_i^2 - (\sum_{i=1}^n f_i)^2][n \sum_{i=1}^n \hat{f}_i^2 - (\sum_{i=1}^n \hat{f}_i)^2]}} \tag{2}$$

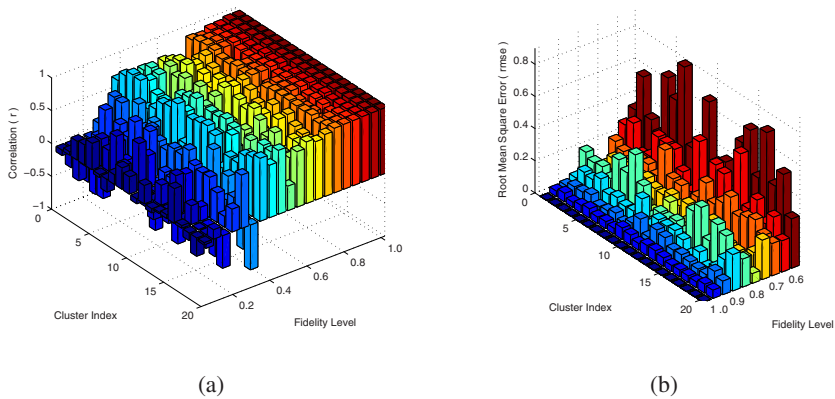


Fig. 2. (a) Correlation factor (r) and (b) root mean square error ($rmse$) of the 20 clustered localized models for variable levels of residual error tolerances, while taking the original airfoil analysis code as the reference model (i.e., with a fidelity of 1)

$$rmse_k^j = \sqrt{\frac{\sum_{i=1}^n |f_i - \hat{f}_i|^2}{n}} \quad (3)$$

where n is the number of data in cluster k , f_i is the output from the original analysis code for data i , and \hat{f}_i is the output from the fidelity level k of data i . The large $rmse$ values in Fig. 2(b) highlighted large discrepancies between the low fidelity models and the original analysis code. On the other hand, Fig. 2(a) displays good fitness landscape correlation between the low fidelity localized models and the original model at early stages of the convergence. In the context of optimization, the absolute fitness of a design is generally non-crucial, rather it is the relative fitness of the solutions that is important in leading the search towards the global optimal solution. In this case, this implies the correlation between the predicted fitness values is of utmost importance [14][15][16].

3 Memetic Framework with Dynamic Fidelity Computational Models

Without loss of generality, we consider computationally expensive general nonlinear programming problems of the form: minimize $f(\mathbf{x})$, subject to $x_i^l \leq x_i \leq x_i^u$, where $i = 1, 2, \dots, d$, d is the dimensionality of the search problem, and x_i^l, x_i^u are the lower and upper bounds of the i^{th} dimension of vector \mathbf{x} , respectively.

Based on previous analysis in Section 2 here we present an evolutionary memetic framework aimed at improving the efficiency of multi-fidelity evolutionary optimization, which is labeled here as Memetic Algorithm with Dynamic Fidelity Models (MA-DFM). The idea behind the proposed MA-DFM is that the complexity level of the localized computational model to be used in place of the exact analysis code, is dynamically determined at run time as the search progresses. In the process, a user specific correlation factor, η , is utilized to determine the minimum level of fidelity that is considered to match well with the original model. The pseudo codes of the standard MA and MA-DFM used in the present work are outlined in Algorithm 1 and 2 respectively.

Algorithm 1. Standard Memetic Algorithm (MA)

- 1: **Initialization:** Generate a population of design vectors.
 - 2: **while** computational budget is not exhausted **do**
 - 3: Evaluate all individuals in the population using the exact fitness function.
 - 4: **for** each individual \mathbf{x} in the population **do**
 - 5: • Apply local search using exact fitness function with probability of P_{ls} and intensity of I_{ls} to find an improved solution, \mathbf{x}_{opt} .
 - 6: • Replace \mathbf{x} with the locally improved solution, i.e. $\mathbf{x} = \mathbf{x}_{opt}$ and $f(\mathbf{x}) = f(\mathbf{x}_{opt})$.
 - 7: **end for**
 - 8: Apply standard evolutionary operators to create a new population.
 - 9: **end while**
-

The MA-DFM (refer to Algorithm 2) begins with the initialization of a population of design points. During the database building phase, the search operates as a traditional evolutionary algorithm for the initial few generations. In this stage, only the original analysis code is used as the fitness function while at the same time, all partial results

encountered along the evaluation of the computational model are archived in database, \mathfrak{S} . Subsequently, the algorithm proceeds to the memetic search phase. Henceforth, for each individual \mathbf{x} , an appropriate fidelity level, φ , of the computational model to use in the local learning phase may then be determined based on the m nearest points in database \mathfrak{S} . For instance, a user-specified confidence based on correlation, η , may be introduced to obtain the minimum fidelity model to be used in place of the original model in the local search. To obtain this minimum fidelity model, we step up the correlation measure starting from the low to high fidelity data found in database \mathfrak{S} , till the fidelity level φ , where $r_\varphi \geq \eta$, is found. Upon local convergence based on the low fidelity model, the locally optimized solution x_{opt} is then validated using the original model and replaces the original starting individual if improved solution is attained, in the spirit of Lamarckian Learning (refer to line 10 of Algorithm 2). The entire search cycle is then repeated until the maximum allowable computational budget is exhausted.

Algorithm 2. MA with Dynamic Fidelity Models (MA-DFM)

```

1: Initialization: Generate a database,  $\mathfrak{S}$  containing a population of designs, archive all evaluations made into the database.
2: while computational budget is not exhausted do
3:   if database building phase then
4:     • Evolve the population using standard EA, archive all evaluations into the database.
5:   else
6:     for each individual  $\mathbf{x}$  in the EA population do
7:       • Find  $m$  nearest points to  $\mathbf{x}$  in database.
8:       • Based on the  $m$  points, find the minimum fidelity level  $\varphi$ , at which the correlation measure between fidelity level of  $\varphi$  and the original analysis code, has reached  $\eta$ , i.e.  $r_\varphi \geq \eta$ 
9:       • Apply local search on  $\mathbf{x}$  using fitness function at fidelity level  $\varphi$ ,  $f_\varphi(\cdot)$  with probability of  $P_{ls}$  and intensity of  $I_{ls}$  to find an improved solution,  $\mathbf{x}_{opt}$ .
10:      • Replace  $\mathbf{x}$  with the locally improved solution, i.e.  $\mathbf{x} = \mathbf{x}_{opt}$  and  $f(\mathbf{x}) = f(\mathbf{x}_{opt})$ .
11:      • Archive all new function evaluations into database  $\mathfrak{S}$ .
12:     end for
13:     Apply standard evolutionary operators to create a new population.
14:   end if
15: end while

```

It is worth highlighting on the novel use of dynamic localized computational model in the proposed framework. Particularly, local models are used in favor of global models since constructing accurate global models is fundamentally flawed due to the curse of dimensionality [16,17]. Further, this allows more precise estimation on the unique characteristics of the problem landscapes, thus leading to the prediction on the appropriate level of localized model fidelity over the use of the original computationally expensive model.

4 Empirical Study

In this section, we present an empirical study using the inverse pressure problem described in Section 2. We validate the efficacy of the proposed MA-DFM against the standard GA, MA, and a non-adaptive MA that employs a fixed fidelity model or MA-FFM in short. In contrast to MA-DFM, note that MA-FFM performs local searches having a pre-defined fixed model of low fidelity (f_ζ), where $\zeta < 1$. The common parametric

Table 1. Setting of experiments for GA, MA, MA-FFM, and MA-DFM

General Parameters	
Population size (N_{pop})	100
Crossover probability (P_{cross})	0.9
Mutation probability (P_{mut})	0.1
Maximum number of exact evaluations	5000
MA, MA-FFM, and MA-DFM - specific Parameters	
Local search probability (P_{ls})	0.2
Local search intensity (I_{ls})	10 local search iterations
MA-FFM and MA-DFM - specific Parameters	
Database building phase	1000 evaluations
Number of nearest neighbours (m)	$(d+1)(d+2)/2$
MA-FFM - specific Parameters	
Fixed lower fidelity level used (ζ)	0.5
MA-DFM - specific Parameters	
Minimum correlation required (η)	0.75

configurations of all 4 schemes used in the present experimental study are summarized in Table 1 and briefly described in what follows.

In the local search procedure, the well-established Feasible Sequential Quadratic Programming (FSQP) method is employed. Although I_{ls} defines the maximum computational budget of each individual in the local learning phase, the actual time incurred do vary according to the fidelity level used. In the present study, the computation cost per evaluation is determined by the fidelity level of the model used, i.e., evaluating a computational model with a fidelity level of 0.7 translates to a compute cost of 0.7 evaluation count. On the other hand, note that each evaluation of the fixed low fidelity in the local search for MA-FFM is performed at a constant cost of ζ , which is here assumed as 0.5 in our experiment setting, which is half the computational expense of the original model. Since model accuracy is highly dependent on the sufficiency of the m data points used for model building, the size of nearest neighboring points used is defined by $(d + 1)(d + 2)/2$, where d is the dimensionality of the optimization problem and is

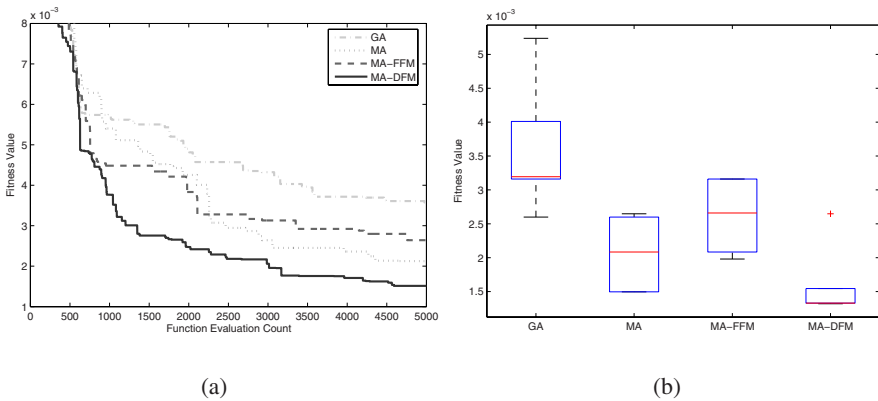


Fig. 3. (a) Search convergence traces of GA, MA, MA-FFM, and MA-DFM on the aerodynamic airfoil problem. (b) Fitness boxplot of GA, MA, MA-FFM, and MA-DFM search at the end of 5000 function evaluations.

Table 2. Mean and standard deviation of GA, MA, MA-FFM, and MA-DFM search at the end of 5000 function evaluations

Scheme	GA	MA	MA-FFM	MA-DFM
Mean±Stdev.	3.5776e-3±9.5973e-4	2.1232e-3±4.9329e-4	2.6417e-3±5.0709e-4	1.5131e-3±4.0862e-4

24 for the airfoil problem considered here. A maximum computational budget of 5000 function evaluations is used in the experimental study.

The obtained average convergence trends for each of the 4 algorithms, across 10 independent runs, are summarized in Fig. 3(a). It is also worth noting that in all the algorithms considered, the first 1000 exact evaluations represent the results of the standard GA, hence similar initial search trends are shown in the figure. Nevertheless, the search trends begin to differ after 1000 evaluations where it can be observed that all the memetic schemes (MA, MA-FFM, and MA-DFM) studied outperform the standard GA significantly. This demonstrates the ability of the MAs in converging to good quality solution more efficiently than a standard GA [18,19].

Next, note that among the two multi-fidelity MA schemes, while both MA-FFM and MA-DFM employs lower fidelity model as the fitness function, the latter is observed to perform better. A *t* – *test* with 95% confidence level for significance of the mean differences on the results based on statistical data from Table 2 and Fig. 3(b), confirms this significant difference in performance.

5 Conclusions

In this paper, we have proposed and studied a multi-fidelity evolutionary optimization technique, which we call the Memetic Algorithm with Dynamic Fidelity Model (MA-DFM). In particular, the evolutionary framework studied is equipped with model fidelity control that decides, at runtime, the appropriate fidelity level of the computational model, which is deemed to be computationally less expensive, to be used in place of the exact analysis code as the search progresses. Empirical study on the aerodynamic airfoil design problem using standard evolutionary optimizers such as GA, MA, and MA-FFM demonstrates that MA-DFM is capable to arrive at improved quality solution and efficiency due to its ability in employing appropriate fidelity level of the computational model in the evolutionary search.

Acknowledgments. D. Lim and Y.S. Ong would like to thank Honda Research Institute Europe GmbH for sponsoring this work and members of Nanyang Technological University, Singapore for providing the computing resources.

References

1. Hutchinson, M.G., Unger, E.R., Mason, W.H., Grossman, B., Haftka, R.T.: Variable-complexity Aerodynamic Optimization of a High Speed Civil Transport Wing. *J. Aircraft* 31, 110–116 (1994)
2. Mason, W.H., Knill, D.L., Giunta, A.A., Grossman, B., Watson, L.T., Haftka, R.T.: Getting the Full Benefits of CFD in Conceptual Design. In: 16th AIAA Applied Aerodynamics Conference, Albuquerque, AIAA-1998-2513 (1998)

3. Raymer, D.P.: *Aircraft Design: A Conceptual Approach*, 3rd edn., Reston, Virginia. Educational Series. AIAA (1999)
4. Ray, T., Tsai, H., Tan, C.: Effects of Solver Fidelity on a Parallel Search Algorithm's Performance for Airfoil Shape Optimization Problems. In: 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization Conference, 2002, Atlanta, Georgia (2002)
5. Keane, A.J.: Wing Optimization Using Design of Experiment, Response Surface, and Data Fusion Methods. *Journal of Aircraft* 40(4), 741–750 (2003)
6. Wu, H.Y., Yang, S., Liu, F., Tsai, H.M.: Comparison of Three Geometric Representations of Airfoils for Aerodynamic Optimization. In: 16th AIAA Computational Fluid Dynamics Conference, Orlando, Florida (2003)
7. El-Beltagy, M.A., Keane, A.J.: A Comparison of Various Optimization Algorithms on a Multilevel Problem. *Engineering Applications of Artificial Intelligence* 12(5), 639–654 (1999)
8. Eby, D., Averill, R.C., Punch III, W.F., Goodman, E.D.: Evaluation of Injection Island GA Performance on Flywheel Design Optimization. In: Parmee, I.C. (ed.) *Adaptive Computing in Design and manufacture*. Springer, Heidelberg (1998)
9. Pantoja, M.F., Meincke, P., Bretones, A.R.: A hybrid genetic-algorithm space-mapping tool for the optimization of antennas. *IEEE Transactions on Antennas and Propagation* 55(3) Part 1, 777–781 (2007)
10. Kampolis, I.C., Zymaris, A.S., Asouti, V.G., Giannakoglou, K.C.: Multilevel optimization strategies based on metamodel-assisted evolutionary algorithms, for computationally expensive problems. In: *IEEE Congress on Evolutionary Computation*, pp. 4116–4123 (2007)
11. Elliott, L., Ingham, D.B., Kyne, A.G., Mera, N.S., Pourkashanian, M., Wilson, C.W.: An Informed Operator Based Genetic Algorithm for Tuning the Reaction Rate Parameters of Chemical Kinetics Mechanisms. In: Deb, K., et al. (eds.) *GECCO 2004*. LNCS, vol. 3103, pp. 945–956. Springer, Heidelberg (2004)
12. Forrester, A.I.J., Bressloff, N.W., Keane, A.J.: Response Surface Model Evolution. In: 16th AIAA Computational Fluid Dynamics Conference, Orlando, Florida, June 23–26 (2003)
13. Hinterding, R., Michalewicz, Z., Eiben, A.E.: Adaptation in Evolutionary Computation: A Survey. In: *IEEE Conference on Evolutionary Computation* (1997)
14. Jin, Y., Huesken, M., Sendhoff, B.: Quality Measures for Approximate Models in Evolutionary Computation. In: *Proc. GECCO Workshops: Workshop on Adaptation, Learning and Approximation in Evolutionary Computation*, Chicago, Illinois, USA, pp. 170–174 (2003)
15. Gräning, L., Jin, Y., Sendhoff, B.: Individual-based Management of Meta-models for Evolutionary Optimization with Applications to Three-dimensional Blade Optimization. In: Yang, S., Ong, Y.-S., Jin, Y. (eds.) *Evolutionary Computation in Dynamic and Uncertain Environments*, pp. 225–250. Springer, Heidelberg (2007)
16. Lim, D., Ong, Y.S., Jin, Y., Sendhoff, B.: A Study on Metamodeling Techniques, Ensembles, and Multi-Surrogates in Evolutionary Computation. In: *Proc. Genetic and Evolutionary Computation Conference*, London, UK, pp. 1288–1295 (2007)
17. Ong, Y.S., Nair, P.B., Keane, A.J.: Evolutionary Optimization of Computationally Expensive Problems via Surrogate Modeling. *Amer. Instit. Aeronaut. Astronaut. J.* 41(4), 687–696 (2003)
18. Ong, Y.S., Keane, A.J.: Meta-Lamarckian Learning in Memetic Algorithm. *IEEE Trans. Evolut. Comput.* 8(2), 99–110 (2004)
19. Ong, Y.S., Lim, M.H., Zhu, N., Wong, K.W.: Classification of Adaptive Memetic Algorithms: A Comparative Study. *IEEE Trans. Syst. Man Cybernet. - Part B* 36(1), 141–152 (2006)

Grid Resource Aggregation Integrated P2P Mode

Zenggang Xiong^{1,2}, Yang Yang², Xuemin Zhang¹, and Ming Zeng²

¹ Department of Computer and Science, Xiaogan University, Xiaogan, P.R. China

² School of Information Engineering, University of Science and Technology Beijing,

Beijing, P.R. China

jkxxzg@163.com

Abstract. As deployed Grids increase from tens to thousands of nodes, Peer-to-Peer(P2P) technologies and protocols can be used to implement scalable services and application in Grid environments. This paper firstly proposes a Grid resource management model integrated P2P mode. Then, in order to allow every user to extract aggregated resources from Grid environments integrated P2P mode, a resource aggregation approach is presented, which is used the multi-agent technology and ant algorithm. Through simulations, we have confirmed that the proposed resource aggregation method can enhance the performance of resource aggregation in Grid environments integrated P2P mode.

Keywords: Grid Computing; Peer-to-Peer Computing; Resource Aggregation.

1 Introduction

During the past several decades, the processing speed of a computer has exponentially increased over one million times. A personal computer today is more powerful than a supercomputer ten years ago. However, today's computer, even the supercomputer, still cannot satisfy the increasing need of life sciences, physics etc. Therefore, the Grid computing and peer-to-peer computing (P2P) obtain development gradually. At present, Grid and Peer-to-Peer are both hot topics respectively.

However, more importantly the convergence of the two systems is increasingly visible: the two research communities started to acknowledge each other by forming multiple research groups that study the potential lessons that exchanged. P2P research focuses more and more on providing infrastructure and diversifying the set of applications; Grid research is starting to pay particular attention to increasing scalability [1]. As Grids used for complex applications increase from tens to thousands of nodes, we should decentralize their functionalities to avoid bottlenecks. The P2P mode could help to ensure Grid scalability: designers can use P2P philosophy and techniques to implement non-hierarchical decentralized Grid systems.

Resource aggregation is the key requirements in large heterogeneous Grid environments, and an effective and efficient resource aggregation mechanism is crucial. Nevertheless, grid resources are potentially very large in number and variety; individual resources are not centrally controlled, and they can enter and leave the grid systems at

any time. For these reasons, resource aggregation in large-scale grids can be very challenging. Traditionally, resource aggregation in grids was mainly based on centralized or hierarchical models. Resource aggregation could be the potential performance and security bottleneck and single point of failure. Using P2P technology, the resource can be aggregation quickly and effectively in Grid environment. scalability and robustness can also be improved in Grid environments integrated P2P mode.

In this paper, we propose a Grid resource management model integrated P2P mode, in which is used multi-agent technology and ant algorithm, to optimize grid resource aggregation. The model overcomes the defects of central resource aggregation mechanism. The ant algorithm is used to aggregate grid resources. The model can adapt to the distributed and dynamic Grid environments, and has a better scalability.

The remainder of this paper is organized as follows: Section 2 gives an introduction to our model of Grid resource aggregation integrated P2P mode. Section 3 describes the resource aggregation algorithm which is used multi-agent and ant algorithm in Grid environments integrated P2P mode. Section 4 gives a simulation result. And finally, section 5 concludes the paper and presents the future work.

2 Grid Resource Management Model Integrated P2P Mode

2.1 Overview of Grid Model Integrated P2P Mode

Generally, the grid model integrated is composed of many supernodes [2]. Each node represents a super management domain. Each node controls the access of a group of local computing resources. It plays two roles: one is as the resource provider, allowing its (or local others) free resources to implement the other supernodes' resources; the other is as a consumer, arbitrarily uses the local resources or the free resources of other supernodes to carry out its task. The idea of grid model integrated P2P mode proposed in this paper comes from [2], but it differs from its idea. As shown in figure 1, the bottom communities of the model using the traditional grid technologies, and the P2P mode is adapted to interact information between GridPeers.

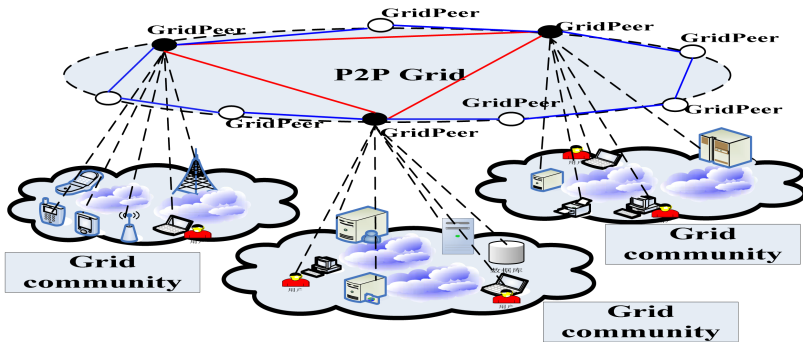


Fig. 1. Overview of Grid model integrated P2P mode

Here, GridPeer is equivalent to a super node. When they search resources, the users firstly query the resources in the domain of GridPeer. If no query result, the search will be carried out through GridPeer to query the other GridPeers with P2P way.

We give some definitions as follows:

Definition 1. $P2PGrid = \{GridPeer_1, GridPeer_2, \dots, GridPeer_n\}$

In which, $GridPeer_1, GridPeer_2, \dots, GridPeer_n$ is $GridCommunity_1, GridCommunity_2, \dots, GridCommunity_n$ corresponding Grid community.

Definition 2. $GridPeer_i = \{Node_1, Node_2, \dots, Node_m\}$

Where, $Node_1, Node_2, \dots, Node_m$ is host or resource with independent ability.

Definition 3. $Node_i = (ID_i, ID_{GridPeer_i})$ represents the attributes of node Node, $ID_i, ID_{GridPeer_i}$ respectively represents node identifier and the GridPeer grid community identifier where Node is.

Definition 4. Membership Function definition—if fuzzy set A located on a regional U, for arbitrary $u \in U$, a designated number $f_A(u) \in [0,1]$, $f_A(u)$ called membership grade of u for A, that is to say, making a mapping $f_A : U \rightarrow [0,1], u \rightarrow f_A(u)$

Membership function can describe ownership degree that nodes on certain types of grid resource. The nodes are classified based on the value of membership function, and classification results are stored in the Grid Community corresponding to GridPeer. The Node selects the corresponding the Grid Community corresponding to GridPeer based on membership function, a Grid Community is composed of one or more common grounds Nodes. Therefore, there will inevitably exist a number of same Nodes between the Grid Community that different GridPeers are in. And super nodes GridPeers are formed by these Nodes. Different Grid Community interact information through GridPeer. Therefore the architecture have a better interconnectivity and better scalability.

2.2 Overview of the Architecture Model of GridPeer

Ian Foster et al. propose the convergence of Grid and agent in [3]. Grid can improve the robust infrastructure of agent system. On the other hand, agent can enhance the autonomous and flexible behavior of grid, too. In a P2PGrid society, each GridPeer can act as an independent and autonomic Grid community. Since dynamic joins and departs of the node and heterogeneous Grid community, and the Grid scale and the above GridPeer overlay topology may change rapidly, it is necessary for each GridPeer to change and configure itself according to the network status. The characteristic of the agent can meet these demands. What's more, multi-agents form an agent organization, which can effectively cooperate with each other to solve problems.

Figure 2 shows the basic components of our architecture. In a P2PGrid society, GridPeer acts as superPeer, the same as it is composed of four agents: RequestAgent, BrokeAgent, TaskAgent and ResourceAgent.

TaskAgent is responsible for creating other agents, and receiving request from users or other GridPeer and transferring the result to the user or GridPeer.

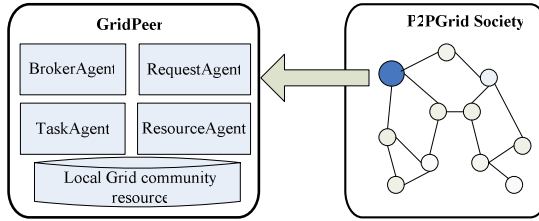


Fig. 2. Overview of the architecture model of GridPeer

RequestAgent is responsible for query intra-GridPeer processing and answer. RequestAgent is also created and dispatched by TaskAgent. When server requests invoke the grid service, it firstly asks for a request from RequestAgent of GridPeer. If the local Grid community resource has the grid service, then RequestAgent returns an answer to TaskAgent. Otherwise, RequestAgent transmits the request information to BrokerAgent.

BrokerAgent is also created by TaskAgent. Its task is that the request from RequestAgent is radiated into another GridPeer and returns the results to RequestAgent, and then RequestAgent transform the results to TaskAgent.

ResourceAgent records and provides the local Grid community resource information and keeps the user's each request results, then the information can be utilized by RequestAgent and BrokerAgent

3 Resource Aggregation Algorithm in Grid Environments Integrated P2P Mode

3.1 Ant Colony Optimization for Optimal Aggregation Tree

Ant colony optimization (ACO) algorithm, one of the most recent techniques for approximate optimization, is a stochastic search procedure. According to Dorigo M, the framework of conventional ACO algorithm is briefly described in [4].

In Grid environments integrated P2P mode, the ant algorithm is runs in two passes. In forward of the algorithm, the route is constructed by one of the ants in which other ants search the nearest point of previous discovered route. The points where multiple ants join are aggregation GridPeers. In the backward pass GridPeers of the discovered path are given weight in form of GridPeer potential which indicates heuristics for reaching to destination GridPeer (for the first ant or other GridPeers) or nearest aggregation point (for other ants) and pheromone trails is the heuristics to communicate other ants of the route discovered. Ants tries to follow the route to get pheromone eventually converges to the optimal route. Non-optimal route pheromone gets evaporated with time. The aggregation points on the optimal tree identify resource aggregation. The indicator in resource aggregation points gives estimate of number of paths aggregates in it.

In Grid environments integrated P2P mode, finding optimal aggregation tree is NP-Hard problem. The reduction is weighted set covering problem. In the ant-based aggregation algorithm, given a permutation of source GridPeers constructs aggregation

tree associated with a cost which is the local best aggregation tree. The algorithm iterates to search the global best and the convergence of algorithm gives the optimal aggregation tree from combinatorial space. Thus, the best aggregation tree constructed by ant routing in iterations is remembered. What's more, giving early aggregation more weight in cost function will converge in optimal aggregation points.

The aggregated traffic in Grid environments integrated P2P mode is local traffic generated by GridPeer and the transit from neighbor GridPeers. In P2P Grid environment, data collected at neighboring GridPeers are often correlated, due to large redundancy in information by multiple GridPeers. In order to reduce communication, the redundant information can be removed without any loss by capturing correlation between GridPeers.

Let denote x_j the random variable measured at GridPeer j . GridPeers without side information coded data with $H(x_j)=R$ bits. However, at intermediate GridPeers on aggregation tree have side information from child GridPeers resulting in reduction on upstream information as $H(x_j|x_i)=r$ bits, $j \neq i$, and $0 \leq r \leq R$.

Therefore, for each edge (i,j) , the data correlation coefficient

$$\text{cost}_{ji_correlation} = H(X_j | X_i) / H(X_j) \quad (1)$$

where $0 \leq \text{cost}_{ji_correlation} \leq 1$, When $\text{cost}_{ji_correlation} = 0$, resources are strongly correlated; when $\text{cost}_{ji_correlation} = 1$, resources are independent.

3.2 Ant Aggregation Algorithm in Grid Environments Integrated P2P Mode

The Grid integrated P2P mode modeled as weighted graph $G(V,E)$, where multiple source $src \in V$ and a single destination $d \in V$. The cost of edge indicates nodes within direct communication range and cost is an estimate measured as Euclidean distance. The input to algorithm is list of source GridPeers $src \in V$. The algorithm assigns ants to source GridPeers. The ants search the routes and communicate with other through pheromones. The GridPeer potential is an estimate of distance to destination. Each ant iterates to construct aggregation tree where internal GridPeers are aggregate points. The ants either try to find shortest route to destination and terminates or finds closest aggregation point of the route searched by previous ants and terminates. The algorithm converges to local best aggregation tree. In order to find the global optimal aggregation GridPeers, the algorithm iterates on different permutations of source GridPeers with cost associated with each to converge in minimum cost. The algorithm converges to minimal cost aggregation points of aggregation tree. In order to converge, with early aggregation, weight is given to search the closest aggregation point. To search shortest path to destination weight is given to the distance estimates to destination. The GridPeer potential remembers estimate closest aggregation point or shortest route to destination. The traced edges are weighted using pheromone which communicates with other ants. The optimal path edges converge with minimal weights of pheromone. The non-optimal edges gets evaporated their pheromone. Thus, the algorithm is governed by pheromone trails and GridPeer potential. The

algorithm runs in two passes: Ants from source moves forward to destination searching the route to destination and ants takes reverse pass from destination to source updating pheromone trails over the edges and updating GridPeer potential as estimate for reaching closest aggregation or destination.

The governing equations for passes are given below:

An artificial ant placed randomly in GridPeers and during each iteration chooses next GridPeer is governed by following formula:

Each ant located at GridPeer i hops to GridPeer j selected among the neighbors that have not yet been visited according to probability. Probability that ant k in GridPeer i will go to GridPeer n :

Select next GridPeer n , namely next hop, from the GridPeer i according to below probability:

$$P_{n,d} = \frac{(T_{n,d}^i)^{\beta_i}}{\sum_{j \in N_d^i} (T_{n,d}^i)^{\beta_i}}, \quad \beta_i \geq 1, \tag{2}$$

Where

$T_{n,d}^i$ is the pheromone value of the entry of routing table T^i for neighbor GridPeer n as the next hop GridPeer to destination node d . N_d^i is the set of neighbors of GridPeer i whose pheromone values are defined for destination GridPeer d , and β_i is a constant. If β_i is large, a neighbor GridPeer with a higher pheromone attracts a reactive forward ant more than in the case with a smaller β_i . In our simulation experiments, β_i is set to one. If no pheromone is available for destination GridPeer d , a forward ant is broadcast to find paths to destination GridPeer d . To eliminate useless ants going toward a wrong direction or taking too long path, the life time of a forward ant is defined by a source GridPeer as the maximum number of hops, i.e., TTL. In addition, when a GridPeer receives several forward ants of the same generation, it decides whether to discard or accept the ants based on the length of the path they travelled.

In the reverse pass each ant updates pheromone trails and GridPeer potential. The pheromone value $T_{n,d}^i$ is updated as follows:

$$T_{n,d}^i \leftarrow \gamma T_{n,d}^i + (1 - \gamma) \left(\frac{T_{n,d}^i + h \cdot T_{hop}}{2} \right)^{-1}, \quad \gamma \in [0, 1], \tag{3}$$

Where $T_{n,d}^i$ is the estimate time from the GridPeer i to destination GridPeer d .

h corresponds to the number of hops from GridPeer i to destination GridPeer n .

T_{hop} is the fixed parameter indicating the time taken for one hop in unloaded conditions.

γ is a smoothing parameter.

The GridPeer potential is weighted function of either reaching nearest aggregation point or shortest route to destination or having high correlation updated as follows:

$$d_{ij} = \gamma^* \cos t_{reaching_aggregation} + \eta^* \cos t_{destination} + k^* \cos t_{ij_correlation} \tag{4}$$

where γ and η are weights of choosing early aggregation or choosing route to destination distance estimates and k is the weight for choosing resource correlation.

4 Simulation Results and Performance Evaluation

We analyze the ant-based resource aggregation algorithm using the Gnutella-like P2P topologies simulator. We randomly choose some existing GridPeers as neighbors for the joining node. We assume the topologies contain $N=100, N=200, N=300, N=400, N=500, N=600$ GridPeers respectively. All the GridPeers are randomly organized in unstructured topology. The number of neighbors of each GridPeer is generated from a normal distribution with mean $N/20$ and standard deviation 1.5. The neighbor GridPeers for every GridPeer are then selected randomly and the connections between GridPeers are set up by initializing the forwarding tables inside each GridPeer. Resources are then simulated on each GridPeer by adding a string identifier corresponding to the name of the resource inside the resource table of a GridPeer. A resource is added on a GridPeer with a probability $p_r = 0.15$ that denotes the availability of the resource in the P2P Grid. GridPeers are also added and removed dynamically as the algorithm runs to simulate the joining and leaving of GridPeers in the P2P Grid. Set of source GridPeer 100, 200, 300, 400, 500, 600 and a destination is considered for generating optimal aggregation tree using ant-based resource aggregation. On varying the ACO parameters and weights of aggregation, shortest distance and correlation, the optimal aggregation tree in P2PGrid is obtained.

The ACO parameters in the simulation setup are given as follows: $\beta_1=2, T_{hop}=3\text{ms}, \gamma=0.7, \eta=0.5, k=0.1$.

We compare greedy method with our ant algorithm. The metrics that used in the comparison is:

Average number of messages: the average number of resource message which reach server GridPeers every 10ms on average in 1000ms time slot, and under the precondition of the same GridPeers and the same resources distribution.

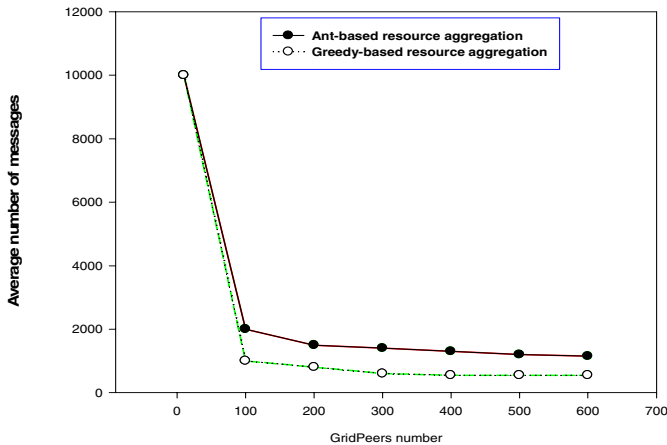


Fig. 3. Average number of messages processed by each GridPeer

On the one hand, Figure 3 shows that the number of messages processed by each server GridPeer can be reduced significantly by increasing the number of server GridPeers, we can see that the load is balanced among server GridPeers uniformly, since the deviation between the maximum and minimum numbers of messages and the average number is not so large in Figure 3. And on the other hand, Figure 3 shows that the system using ant-based resource aggregation algorithm performs better than the system with greedy resource aggregation algorithm in terms of the average number of message. With the increase of the GridPeers, the ant-based resource aggregation algorithm outperform the greedy resource aggregation.

5 Conclusions and Future Work

In this paper, we firstly propose a Grid resource management model integrated P2P mode, then we apply the multi-agent technology and ant algorithm to aggregate the grid resource. In the future work, we plan to further optimize our algorithm and compare our model with other resource aggregation methods.

Acknowledgments. This research is partially supported by National Natural Science Foundation in China (No.60673160), Xiaogan University Natural Science Foundation (Z2007039).

References

1. Ian, F.A.I.: On Death, Taxes, and the Convergence of Peer-to-Peer and Grid Computing. In: Kaashoek, M.F., Stoica, I. (eds.) IPTPS 2003. LNCS, vol. 2735, pp. 118–128. Springer, Heidelberg (2003)
2. Li, X., Zhuang, Z.Y., Liu, Y.H.: Dynamic Layer Management in Super peer Architectures. *IEEE Transactions on Parallel and Distributed Systems* 16(11), 1078–1091 (2005)
3. Ian, F., Nicholas, J., Kesselman, C.: Brain Meets Brawn: Why Grid and Agents Need Each Other. In: *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pp. 8–15. ACM Press, New York (2004)
4. Dorigo, M., Maniezzo, V., Colomi, A.: The Ant System: Optimization by A Colony of Co-operating Agents. *IEEE Trans. On Systems, Man and Cybernetics, Part-B* 26(1), 29–41 (1996)

Hybrid Scatter Search with Extremal Optimization for Solving the Capacitated Vehicle Routing Problem

Kai Sun, Gen-Ke Yang, and Yu-Wang Chen

Department of Automation, Shanghai Jiaotong University, 800, Dongchuan Road,
Shanghai, China. P.R.
{sunKai, gkyang, cywpeak}@sjtu.edu.cn

Abstract. Capacitated vehicle routing problem (CVRP) is one of the most well-known and well-studied scheduling problems in operation research and logistics. However, it is quite difficult to achieve an optimal solution with the traditional optimization methods owing to the high computational complexity. The paper focuses on developing a hybrid optimization algorithm by incorporating the extremal optimization into a scatter search conceptual framework to solve the CVRP. The proposed algorithm is applied to a set of benchmark problems and the performance of the algorithm is evaluated by comparing the obtained results with the results published in the literature. Comparing results indicate that this new method is an effective and competitive approach for the capacitated vehicle routing problem.

Keywords: Capacitated vehicle routing problem; Scatter search; Extremal optimization.

1 Introduction

The vehicle routing problem (VRP), introduced by Dantzig and Ramser [1], is a well-known combinatorial optimization problem in the field of service operations management and logistics. The capacitated vehicle routing problem (CVRP) is a NP-hard problem for simultaneously determining the routes for several vehicles from a central depot to a set of customers, and then returning to the depot without exceeding the capacity constraints of each vehicle. The objective is to determine a viable route schedule which minimizes the distance or the total cost with various constraints.

A variety of algorithms including exact methods [2] and efficient heuristics have been developed to solve the CVRP. Since the CVRP is NP-hard, no exact algorithm can consistently solve CVRP-instances with more than 50 customers. Therefore, most of the research has focused on developing meta-heuristics to produce high quality solutions in a limited time. Available meta-heuristics include simulated annealing algorithms [3], tabu search [4], genetic algorithms [5], and ant colony algorithm [6], etc. However, each meta-heuristic has its own strength and weakness. Therefore, much research has tried to develop hybrid algorithms expecting to achieve complementarity, which improves the effectiveness and efficiency. Some previous

experiments have showed that hybrid algorithms have better effectiveness and efficiency than those of simplistic ones [7][8].

In the paper, we introduce a very effective hybrid method based on scatter search (SS) and extremal optimization (EO) to solve the problem. The new hybrid optimization algorithm combines the strong global search ability of SS and the strong local search ability of EO, and computational experiments demonstrate that this new method is very effective and competitive for solving the capacitated vehicle routing problem.

The rest of the paper is organized as follows: Section 2 describes the general SS and EO algorithm, and how to incorporate the EO into a SS conceptual framework to solve the CVRP. Section 3 presents the computational results and comparisons. Some concluding remarks are given in section 4.

2 Hybrid SS-EO Algorithm for CVRP

Scatter search (SS) is an evolutionary method that has been successfully applied to hard optimization problems [9][10]. Unlike genetic algorithm, a scatter search operates on a small set of solutions (called reference set) and makes only limited use of randomization as a proxy for diversification when searching for a globally optimal solution. SS consists of five components which are diversification generation method, improvement method, reference set update method, subset generation method and solution combination method. A template of the standard SS can be shown as follows:

Step 1: Initialization

- 1.1 Use **diversification generation method** to generate diverse trial solutions.
- 1.2 Apply the **improvement method** to enhanced trial solutions.
- 1.3 Apply the **reference update method** to build the initial reference set (*RefSet*) from the enhanced trial solutions.

Step 2: Computation

Do

- 2.1 Generate subsets of *RefSet* with the **subset generation method**.
- 2.2 Use **solution combination method** to combine these subsets and obtain new solutions.
- 2.3 Use the **improvement method** to enhance these new trial solutions.
- 2.4 Apply the **reference update method** to update the *RefSet*.

While (the maximum generation is not meet)

Step 3: Output the optimization results.

2.1 Solution Representation and Fitness Function

In the paper, we use a permutation of $N + K$ integers to encode the antibody, where there are N customers to be served by K vehicles. The integers from 1 to N represent the customers to be served, and the integers $N + 1, N + 2, \dots, N + K$ represent K dummy nodes that separate different routes in the whole CVRP. With this representation, the information on both the stations and the trucks are embedded in a single solution structure.

Fitness is used to evaluate the performance of solutions, and choosing a proper objective function as fitness function to represent the corresponding quality of each solution is very important for the SS algorithm. In the paper, we take the total distance traveled as the fitness function.

2.2 Diversification Generation Method

The diversification generation method is used to generate a collection of diverse trial solutions, using an arbitrary trial solution (or seed solution) as an input. This element of the SS approach is particularly important, given the goal of developing a method that balances diversification and intensification in the search. This method was suggested by Glover [9], which generates diversified permutations in a systematic way without reference to the objective function.

2.3 Improvement Method

Each of the new trial solutions which are obtained from the diversification generation method or solution combination method is subjected to the improvement method. This method aims to enhance the quality of these solutions. In the paper, we take two versions of local search meta-heuristics to improve trial solutions. A long-term EO-based improvement method is only applied to the best new trial solution, and a short-term 2-opt local search is taken to enhance other new trial solutions. With the hybridization of these two local methods, it can get a compromise between solution quality and computational effort. For more detail of the 2-opt local search can be, please refer to [15].

EO is a new stochastic method which origins from the model of self-organized criticality (SOC) in ecosystems [11]. Evolution in this model is driven by a process where the weakest species in the population, together with its nearest neighbors is always forced to mutate. Boettcher and Percus [12] have adapted the evolutionary model of [11] to tackle hard problems in combinatorial optimization, calling their algorithm extremal optimization. The computational flow of EO can be described as follows:

Step 1: Initialize a configuration S , and set $S_{best} = S$.

Step 2: For the “current” configuration S

- (a) Evaluate localized fitness λ_i for each variable x_i ,
- (b) Find j with $\lambda_j \geq \lambda_i$ for all i , i.e., x_j has the worst localized fitness,
- (c) Choose a $S' \in N(S)$ such that the “worst” x_j must change its state,
- (d) If $C(S') < C(S_{best})$, then set $S_{best} = S'$,
- (e) Accept $S \leftarrow S'$ unconditionally.

Step 3: Repeat at step 2 as long as desired.

Step 4: Return S_{best} and $C(S_{best})$.

To improve performance and avoid a search based only on the forced mutation of the weakest species, they modified the basic EO algorithm introducing an adjustable

parameter so that the search could escape local optima [13]. This variation of the EO algorithm is called τ -EO and showed superior performance over the basic EO.

In order to apply the τ -EO algorithm to solve the CVRP, we have given several new definitions on the species and localized fitness of the problem. The detailed discussion of applying EO algorithm is shown as follows.

2.3.1 Species and Localized Fitness

In the EO method, each variable is considered as one species and to each of them a localized fitness is assigned that represents the level of adaptability of that species. To solve the CVRP by the EO algorithm, each customer or dummy node (depot) of a solution can be seen as a node (species) of the CVRP tour, and then a localized fitness is assigned to each node.

Let $D = \{d_{ij}\}_{(N+1) \times (N+1)}$ be the distance matrix, where d_{ij} ($i = 1, \dots, N, j = 1, \dots, N$) is the inter-customer distance between customer i and customer j , and $d_{i(N+1)}$ or $d_{(N+1)i}$ ($i = 1, \dots, N$) represents the distance between customer i and the depot. Assuming p_i to be the length of the edge starting from node i in a CVRP tour, the localized fitness of node i is defined as:

$$\lambda_i = \begin{cases} p_i - \min_{j \neq i} (d_{ij}) & \text{for each node } i \leq N \\ p_i - \min_{j \neq N+1} (d_{(N+1)j}) & \text{for each node } i > N \end{cases} \quad (1)$$

2.3.2 EO Algorithm for CVRP

Based on the definition of species and localized fitness, the improvement method with EO is proposed as follows:

(1) The optimization process starts from a new trial solution obtained from the diversification generation method or solution combination method. Let S_{best} represent the best solution found so far, we set $S_{best} \leftarrow S$ initially.

(2) To the current solution S , the localized fitness λ_i for each node ($1 \leq i \leq N + K$) can be calculated by Equation (1). In our algorithm, the solution components (nodes) are ranked according to their localized fitness, using a rank $r = 1$ for the worst nodes and $r = N + K$ for the best. A candidate node i is chosen for mutation and confirmed with a probability:

$$P(i) = r^{-\tau} \quad (2)$$

where r is the localized fitness rank of node i , and the power-law exponent τ is an adjustable parameter. How to choose a proper τ is very important in the application of EO. Boettcher [13] demonstrates that a value $\tau \sim 1 + 1/\ln(n)$ seems to work best by numerous experiments, where n is the number of species in the solution.

(3) For updating the state of the selected nodes, we construct the neighborhood $N(S)$ of solution S by the 2-opt move [15]: construct new Hamilton cycles by deleting two edges and reconnecting the two resulting paths in a different way. Since

the localized fitness of the selected nodes is expected to be updated, the 2-opt move must replace its forward-directed edge, and so there will be $N + K - 1$ possible solutions. Being in the current state S , the random walker in our optimization dynamics chooses a new state S' which has μ th ‘good’ solution quality of its neighbors $N(S)$ by another power-law:

$$P_{(\mu)} = \mu^{-\alpha} \quad (1 \leq \mu \leq |N(S)|) \tag{3}$$

Then accept $S \leftarrow S'$ unconditionally, if $C(S') < C(S_{best})$, set $S_{best} = S'$, where $C(S')$ is the total distance of the solution S' .

(4) Repeat the update step (2) until a given termination criterion is satisfied, which is either a certain number of iterations or a predefined amount of CPU time.

(5) Return S_{best} and $C(S_{best})$ as the results, and S_{best} is an improved trial solution for SS.

2.4 Reference Update Method

The reference set is a subset of both diverse and high-quality trial solutions. In the paper, the reference set, *RefSet*, consists of two subsets, *RefSet₁* and *Refset₂*, of maximum size b_1 and b_2 , respectively. The first subset contains the best quality solutions, while the second subset should be filled with solutions promoting diversity.

Assume that the population of improved trial solutions at each generation is *Pop*. The construction of the initial *RefSet₁* starts with the selection of best b_1 solutions from *Pop*. These solutions are added to the *RefSet₁* and deleted from *Pop*.

After update the *RefSet₁*, the b_2 diverse solutions which are determined by distance between solutions (δ) are included in the *RefSet₂*. The δ can be calculated as follows: assume there are two solutions S and S' , the distance for these two solutions $\delta(S, S')$ can be obtained by counting the number of the same edges between them. For each solution S in *non-RefSet*, the maximum δ , $\delta_{max}(S)$ to the solutions in *RefSet* is calculated:

$$\delta_{max}(S) = \max_{S' \in RefSet} \{\delta(S, S')\} \tag{4}$$

The b_2 solutions with the minimum $\delta_{max}(S)$ is added to the *RefSet₂* and deleted from *Pop*.

2.5 Subset Generation Method

The subsets generation method consists of generating the subsets that will be used for creating new solutions with the solution combined method. In the paper, the method is organized to generate three different types of 2-elements subsets, which are shown as R_1, R_2, R_3 . The implementation of the subsets is explained as follows:

R_1 : Generated by pairs of solutions from the *Refset₁*.

R_2 : Generated by pairs of solutions from the *Refset₂*.

R_3 : The first element of i th subset is the i th solution in the $Refset_1$ and the second element is the i th solution of the $Refset_2$.

2.6 Solution CVombination Method

This method uses each generated subset and combines the subset solutions, returning one or more trial solutions. In the paper, we take Insert-based crossover (IB_X) operator [5] as solution combination method.

3 Computational Results and Comparison

To illustrate the effectiveness of the hybrid method, a computational experiment was conducted on the instances from Vehicle Routing Data Sets (<http://www.Branchandcut.org/VRP/data/>) and the results were compared with those from GA with 2-opt [7] and those from DPSO-SA [8]. The algorithms are programmed in Borland C++ and run it on the Intel Pentium 2.4G with 256M RAM. The parameters values of SS-EO are defined as follows: maximum number of generations, population size and reference set size b_1, b_2 are set according to the problem scale, parameter τ of EO is set to $1+1/\ln(N+K)$ and parameter β of EO is 3. EO sub-program was terminated whenever there was no improvement in 20 successive generations which enables a reduction in running time. Each instance is randomly performed 20 times for each algorithm.

In order to illustrate the performance of EO, we design a hybrid SS-SA algorithm by incorporating SA-based local search into the SS template. Table 1 presents the computational performance of SS-EO and SS-SA. BKS is the best known solution so far, C^* is the best solution found by each algorithm, % is the percentage of average objective value of algorithms over BKS , and t is average CPU time on the Intel Pentium 2.4G with 256M RAM. It can be seen that the solution quality and running time of SS-EO are better than those of SS-SA, which shows that the EO we developed is a better local search method for solving the CVRP than SA. The hybrid SS-EO method can find BKS or near- BKS for many problem instances in a reasonable

Table 1. Comparison results of SS-SA and SS-EO

Problem	BKS	SS-SA			SS-EO		
		C^*	%	t	C^*	%	t
A-n33-k5	661	661	0	11.4	661	0	8.5
A-n46-k7	914	914	0.91	53.9	914	0.65	32.5
A-n60-k9	1354	1354	0.85	85.8	1354	0.76	61.4
B-n35-k5	955	955	0.45	17.4	955	0.21	13.0
B-n45-k5	751	751	0.92	36.9	751	0.78	29.7
E-n51-k5	521	535	1.12	108.6	521	0.75	81.4
E-n76-k7	682	688	1.56	185.7	682	1.35	149.8
M-n101-k10	820	824	1.78	295.2	824	1.45	214.8
P-n76-k4	593	598	2.32	165.8	598	1.56	109.3
P-n101-k4	681	694	1.75	689.2	688	1.43	471.9

amount of computing time, which demonstrates the effectiveness and the global search property of the hybrid search, and the superiority of the average performance over 20 random runs shows that the hybrid optimization algorithm is very robust.

Table 2 shows the comparison of SS-EO with other well-known algorithms from literature. The column labeled GA with 2-opt refers to the algorithm [7] and the DPSO-SA is the algorithm by Chen et.al [8]. From table 2, we can see that the solution quality of SS-EO is better than those got from GA with 2-opt and DPSO-SA, which demonstrates that the hybrid SS-EO algorithm is a competitive and effective approach for solving CVRP.

Table 2. Comparison of SS-EO with other algorithm

Problem	N	K	BKS	GA with 2-opt	DPSO-SA	SS-EO
A-n33-k5	32	5	661	661	661	661
A-n46-k7	45	7	914	928	914	914
A-n60-k9	59	9	1354	1360	1354	1354
B-n35-k5	34	5	955	955	955	955
B-n45-k5	44	5	751	762	751	751
E-n51-k5	50	5	521	531	528	521
E-n76-k7	75	7	682	697	688	682
M-n101-k10	100	10	820	836	824	824
P-n76-k4	75	4	593	605	602	598
P-n101-k4	100	4	681	706	694	688

4 Conclusion

We have developed a new hybrid method by combining scatter search with extremal optimization to solve the capacitated vehicle routing problem. This hybrid method combines the advantages of two algorithms and mitigates the disadvantages of them. The performance of the SS-EO algorithm is evaluated in comparison with results obtained from other authors' algorithms for a number of benchmark instances, and the results show that the new algorithm is very effective and efficient. It can find optima for most test problems and has high quality solutions for almost all test problems in a reasonable running time. The proposed approach is also suitable for other problems, especially suitable for discrete optimization problems. Research on applying the proposed approach to other combinatorial optimization problems should be an interesting subject.

Acknowledgments. This research is partially supported by the National Nature Science Foundation of China (No. 60574063).

References

1. Dantzig, G.B., Ramser, R.H.: The Truck Dispatching Problem. *Management Science* 10, 80–91 (1959)
2. Christofides, N., Mignozzi, A., Toth, P.: Exact Algorithms for the Vehicle Routing Problem Based on Spanning Tree and Shortest Path Relaxations. *Math. Pto.* 20, 255–282 (1981)

3. Osman, I.H.: Meta-strategy Simulated Annealing and Tabu Search Algorithms for the Vehicle Routing Problem. *Annals of Operations Research* 41, 421–451 (1993)
4. Brandao, J., Eglese, R.: A Deterministic Tabu Search Algorithm for the Capacitated Arc Routing Problem. *Computers & Operations Research* 35, 1112–1126 (2008)
5. Berger, J., Barkaoui, M.: A Hybrid Genetic Algorithm for the Capacitated Vehicle Routing Problem. *Journal of Operation Research Society* 54, 1254–1262 (2003)
6. Mazzeo, S., Loiseau, I.: An Ant Colony Algorithm for the Capacitated Vehicle Routing. *Electronic Notes in Discrete Mathematics* 18, 181–186 (2004)
7. Wang, Z.Z.: A Hybrid Optimization Algorithm Solving Vehicle Routing Problems. *Operations Research and Management Science* 13, 48–52 (2004)
8. Chen, A.L., Yang, G.K., Wu, Z.M.: Hybrid Discrete Particle Swarm Optimization Algorithm for Capacitated Vehicle Routing Problem. *Journal of Zhejiang University Science A* 7, 607–614 (2006)
9. Glover, F.: Scatter Search and Path Relinking. In: Corne, D., Dorigo, M., Glover, F. (eds.) *New Ideas in Optimization*, pp. 297–316. McGraw-Hill, New York (1999)
10. Noorul, A., Saravanan, M.: A Scatter Search Approach for General Flowshop Scheduling Problem. *International Journal of Advanced Manufacturing Technology* 31, 731–736 (2007)
11. Bak, P., Kan, C.: Self-organized Criticality. *Scientific American* 264, 26–33 (1991)
12. Boettcher, S.: Extremal Optimization: Heuristics via Coevolutionary Avalanches. *Computing in Science Engineering* 2, 75–82 (2000)
13. Boettcher, S., Percus, A.G.: Optimization with Extremal Dynamics. *Complexity* 8, 57–62 (2003)
14. Chen, Y.W., Lu, Y.Z., Chen, P.: Optimization with Extremal Dynamics for the Traveling Salesman Problem. *Physica A: Statistical Mechanics and its Applications* 385, 115–123 (2007)
15. Helsgaun, K.: An Effective Implementation of the Lin-kernighan Traveling Salesman Heuristic. *European Journal of Operational Research* 126, 106–130 (2000)

Implement Web Learning System Based on Genetic Algorithm and Pervasive Agent Ontology

Qinglin Guo^{1,2} and Ming Zhang²

¹ School of Computer Science and Technology, North China Electric Power University,
Beijing 102206, China
qlguo88@sohu.com

² Department of Computer Science and Technology, Peking University,
Beijing 100871, China
{qlguo,mz88}@pku.edu.cn

Abstract. For a web-based dynamic learning environment, personalized support for learners becomes more important. In order to achieve optimal efficiency in a learning process, individual learner's cognitive learning style should be taken into account. It is necessary to provide learners with an individualized learning support system. In this paper, a framework of web learning system based on genetic algorithm and Pervasive Agent Ontology is presented. The proposed framework utilizes genetic algorithm for representing and extracting a dynamic learning process and learning pattern to support students' deep learning in web-based learning environment. Aiming at the problems in current Web environment, we put forward the information integration method of Semantic Web based on Pervasive Agent Ontology (SWPAO method), which will integrate, analyze and process enormous web information and extract answers for students on the basis of semantics. And experiments do prove that it is feasible to use the method to develop an individual Web-based learning system, which is valuable for further study in more depth.

1 Introduction

The need for providing learners with web-based learning content that match their accessibility needs and preferences, as well as providing ways to match learning content to user's devices has been identified as an important issue in accessible educational environment. For a web-based open and dynamic learning environment, personalized support for learners becomes more important [1]. In order to achieve optimal efficiency in a learning process, individual learner's cognitive learning style should be taken into account. Due to different types of learners, it is necessary to provide them with an individualized learning support system.

With the popularization of the Internet and increasingly diverse online information, some search engines such as Google, Yahoo, Baidu have gradually become a part of people's life. However the traditional search engine has many limitations, for instance, it returns a lot of relative web pages rather than accurate answers. In addition, it only regards keywords as index without relating to semantic information, making it

difficult to really understand what the user intends to do. In the Web learning system, users could use sentences in daily life to raise questions and the system will return answers to users directly after analyzing and comprehending these questions. Therefore, the Web learning system better satisfies the requirements of users. The data in web has the features of semi-structure, isomerism and distributivity. Blocking these features and providing uniform model for users are the key questions of web information integration. However, the inherent features of isomerism, distributivity, growth and variation of the data in internet determine that the structural method is not adaptable to web information integration. With the vigorously generalization of semantic web by W3C, semantics oriented web information integrating method has been the major point of the research on web information integration technology [2]. One common semantic model must be offered to solve the problem of semantic isomerism in semantics oriented web information integration, which is a platform independent model to block semantic isomerism among web information. As “the set of conceptions and relations among conceptions in specific domain” [3], ontology could efficiently express the general knowledge in specific domain, which is adaptable to be the common semantic model of semantics oriented web information integration.

In recent years, ontology arouses widespread concern in the artificial intelligence domain and it has been applied extensively [4]. Adopting ontology knowledge base in QA could better indicate the inherent relations among knowledge, rationalize knowledge organizations, reduce redundant memories and help to extract answers based on semantics. In many web information integration applications which use enquiry-driven integration mode [5] facing to information sources of multiple domains [6], the enquiries of users usually cover many information sources. Therefore, users usually expect to search their interested information based on only one special ontology without caring for answers received from which information sources or through what kind of processing. To meet this demand, a general ontology should be constructed based on local domain ontology, which is called Pervasive Agent Ontology (PAO for short). Pervasive Agent Ontology and local ontology should operate mutually through shared lexical collection, and they should be of mapping relations. On such basis, we put forward the information integrated method of Semantic Web based on Pervasive Agent Ontology, SWPAO method for short.

2 System Components Architecture

As a contribution to the former Systems Components layer 3, five processes are identified instead of four in the original specification: learner entity, evaluation, coach, delivery process, and University. Also, there are two repositories: learner records and learning resources. First of all, the university process is considered because it biases onto educational system in a direct way. Wherefore, the coach process has been divided into two tasks: coach and virtual coach, because this process has to adapt to the learners' individual needs. During the learning process some decisions about sequence, quizzes, and activities can be manually made for the coach and others can be automatically fulfilled for the virtual coach.

Briefly, the overall operation has the following form: (1) Learner information assessment. This new operation collects information about the behavior of the student

along the course, e.g., trajectory, spending time, nomadicity, and navigation. This information is stored into learner records via components. (2) Evaluation of the learner based upon multimedia interactions context. (3) Elicitation of assessments and learner information stemmed from the evaluations. (4) Personalized learning pattern based on genetic algorithm. (5) Organization of a learner history database regarding to: keyboard clicks, mouse clicks, voice responses, choices, written responses, and many details more. (6) Query support to the coach in order to review the learner's assessment and learner information, such as: preferences, past performance history, and future learning objectives. (7) Information integration of Semantic Web based on Pervasive Agent Ontology. (8) Seeking learning resources, via query and catalog info, for appropriate learning content demanded by coach and virtual coach. (9) Extraction of locators from the available catalog info by Coach and virtual coach in order to send them to the delivery process, e.g., a lesson plan or references to contents. (10) Answer extraction based on Pervasive Agent Ontology.

3 Personalized Learning Pattern Based on Genetic Algorithm

Generated courseware tailor a courseware to each learner based on answers to a pre-test before at the start of the learning course unit. Generated coursewares are helpful to individual learners for performing more efficient learning specially when learners have different needs, varying desires, and different levels of knowledge background. The details of the proposed genetic-based curriculum sequence scheme are presented in the next subsection.

Genetic algorithm only needs to appraise chromosome produced by the system, choose the chromosome by adaptive value, and then give more multiplication chances for more adaptive chromosome. The theory basis of genetic algorithm is binary expression of genetic algorithm and implication of the pattern. The pattern is the template to explain the comparability between chromosomes [7]. Let's suppose that at the time step t , a specific mode s contains m origin parameter in group $P(t)$. Firstly, we don't consider crossing and variant operation. According to the adaptive value of the cause parameter, every origin parameter gets different duplication probability. The duplication probability of the origin parameter i is :

$$p_i = \frac{f(i)}{\sum_{j=1}^n f(j)} \quad (1)$$

Then in the group $P(t+1)$, the expected value of the mode s is:

$$E[m(s, t+1)] = m(s, t) \cdot n \cdot \frac{\bar{f}(s)}{\sum_{j=1}^n f(j)} \quad (2)$$

Among them, $\bar{f}(s)$ indicating the average value of all origin parameter about mode s at the time t .

Suppose, the average value of all individual adaptation value about $P(t)$ is:

$$\bar{f} = \frac{\sum_{j=1}^n f(j)}{n} \tag{3}$$

Then formula (2) may be expressed as:

$$E[m(s, t + 1)] = m(s, t) \cdot \frac{\bar{f}(s)}{\bar{f}} \tag{4}$$

If the adaptive value of the mode is $(1 + c)\bar{f}$, in which c is a constant, then formula (4) can be:

$$\begin{aligned} E[m(s, t + 1)] &= m(s, t) \cdot \frac{(1 + c)\bar{f}}{\bar{f}} \\ &= m(s, t) \cdot (1 + c) = m(s, 0) \cdot (1 + c)^{t+1} \end{aligned} \tag{5}$$

Formula (5) indicates that above the average adaptation value, the pattern can be duplicated. The duplication result has not produced the new mode. Thus, we need to use crossed and variant operation. Let's suppose that the nondestructive probability of mode s is p , then:

$$p_s \geq 1 - \frac{\delta(s)}{l - 1} \tag{6}$$

If the crossed probability is p_c , then the probability that s can't be destroyed is :

$$p_s \geq 1 - p_c \cdot \frac{\delta(s)}{l - 1} \tag{7}$$

Therefore, formula (4) may be represented as:

$$E[m(s, t + 1)] = m(s, t) \cdot \frac{\bar{f}(s)}{\bar{f}} \left[1 - p_c \cdot \frac{\delta(s)}{l - 1} \right] \tag{8}$$

In the proposed mutation operation, two randomly selected genes (learning pattern) in an individual are forced to exchange the gene to each other under probability decision. The proposed mutation operation is similar to the mutation operation of swapping two-points implemented in the standard genetic algorithm. The only difference is that the binary-coded scheme is employed in the standard genetic algorithm, but the integer-coded scheme was employed to represent an individual string, i.e. a potential solution for the genetic algorithm, in the study. This scheme can avoid generating illegal learning paths mentioned in the previous subsection. The mutation operation can evolve some new individuals that might not be produced by the operations of reproduction and crossover to avoid that the solution traps into the local optimum. Generally, a low probability of mutation can guarantee the convergence of genetic algorithm, but it may lead to poor quality solution. By contrast, a high probability of

mutation may lead to the phenomenon of random walk in the genetic algorithm, thus reducing convergence speed.

The extracted learning pattern needs to be sheared. The aim of applying genetic algorithm for learning pattern shearing is to select a small amount of unique learning pattern from the mass learning pattern extracted from the former stage. The group is randomly initialized by the learning pattern base extracted from the former stage. At first, a group composed of N chromosome sets is generated. Every chromosome string is assessed and classified by the degree of matching at the same time. A half chromosome strings with the highest the degrees of matching are chosen as parents in current group. Then the child string is generated from their parents using one point interaction and the child string has the variation application to keep variation in the last chromosome. The genetic operation is repeatedly operated on the different subsequent generations.

We adopt the genetic algorithm to extract learning pattern from the system. We also compared these learning patterns extracted by this method with the learning patterns from original neural network [8] and C4.5 decision tree [9]. The comparison shows that the method is effective. In this paper we use the algorithm to mine the learning pattern of every student. The example sets of this question contain 160 examples. In every experiment, we used test set to test the learning pattern set extracted by C4.5 decision tree, original neural network, and the learning pattern set extracted from the genetic algorithm. The results are listed in Table 1. In this table, the “fidelity of the method” represents the proportion of testing examples in the testing set, and the testing examples are those which having the same results for the original neural network and the extracting method based on genetic algorithm.

Table 1. Comparing experimental result of different method

NO.	precision			fidelity of the method (%)
	C4.5(%)	original neural network (%)	the method in this paper (%)	
1	80.5	91.2	95.7	96.7
2	76.3	85.8	89.1	93.6
3	72.8	93.1	86.5	90.4
4	83.6	95.4	97.9	96.3
average	78.3	91.4	92.3	94.2

From Table 1, the mode of neural network has higher precision, because the training set offered to neural network and C4.5 decision tree has noise, whereas the algorithm in this paper has good ability of generalization and noise tolerance.

4 Answer Extraction Based on Pervasive Agent Ontology

After analyzing the questions presented by users, some entities and their attributes or entity-entity pair will be received, and these results will be submitted to answer extraction module. First of all, the entity is mapped on the corresponding catalogue in class hierarchy of ontology knowledge base. In the following, find out this entity in

all corresponding entities of this catalogue, and then search the attribute value needed. If corresponding catalogue is searched instead of the entity, it is necessary to trace back to father catalogue and find out whether having the attribute or not. If having it, the attribute value should be returned. If not, it is necessary to trace back again until to the direct subclass of abstract class. As to the <entity, entity> which shows the relation of conceptions, we have to find out whether its corresponding catalogues have relations. If existing, relation value should be returned. Otherwise, we have to search whether father catalogues have relations. If having, relation value should be returned. If not, it is necessary to continue tracing back until to the common ancestor class.

We resort to SPARQL language and Jena inference engine to search for answers. Jena inference engine provides perfect support for ontology modeling, operation and inference and other relative activities. SPARQL language is an ontology searching language recommended by W3C, which uses the searching form of SQL sentences. SPARQL language offers the function of ontology enquiry. In ontology, all knowledge is in form of triplet (<subject, property, object>). SPARQL only searches the triplet saved in ontology library, without the function of inference. Therefore, we embed SPARQL enquiry into Jena inference engine to get implied triplet by inference, and then return the answers to users. The inference of Jena is based on rules. Its configurations include OWL ontology inference engine OWLReasoner and RDFS ontology inference engine RDFSReasoner, etc, and its inference rule is implicated in the rule documents which confine attributes and relations. Jena supports forward engine inference, backward engine inference and promiscuous mode inference. Under promiscuous mode, forward engine use forward rules to carry on matching inference on raw data. The data inferred will be stored in an internal deductive database, and the forward rules which assert new backward rules will deliver the rules deducted by the rules tied by forward variable to backward engine. Then, backward engine will adopt the combination of initial forward rules and new generated rules to apply on raw and deductive data in order to put out final results.

The advantage of adopting promiscuous inference mode is that backward rules only work on current relative data set which will obtain greater performance. For example, SubPropertyOf restriction implemented by using RDFS rules in one inference could use the following rules: [(? a? q? b) <-(?p Rdfs:subPropertyOf ?q), (?a ?p ?b)]. Every target should be matched with the head of the rule, so every enquiry will activate a dynamic test: whether sub-attribute of the current attribute search is existed. Oppositely, mixing rules: [(?p Rdfs:subPropertyOf ?q), Not Equal(?p,?q)->[(?a ?q ?b)<-(?a ?p ?b)]] will preprocess all sub-attribute relations and store them in a simple rules chain, which will be activated only by finding out an attribute with sub-attribute. If not, there is no enquiry time of head for this rule.

We resort to OWLReasoner to alter and redefine the needed rules on the basis of Jena's original rules document, and to infer by promiscuous mode. Figure 1 is one example of ontology part, which illustrates that a triplet exists in ontology base <male disease consulting center, treat, acute proctitis >, <Professor Wang, belongto, male

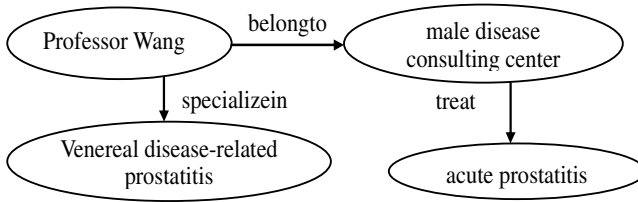


Fig. 1. Example of inference in ontology

disease consulting center> and <Professor Wang, specializein, venereal disease-related prostatitis>. However, the triplet <Professor Wang, specializein, acute prostatitis > is not existed in ontology base, which will be received by inference.

The following rules will be defined in rules document:

[rule1: (?A pre: treat? C), (? B pre: belongto? A) -> (? B pre: Specializein? C)]

The experimental results are:

Professor Wang specialize in acute prostatitis

Professor Wang specialize in venereal disease-related prostatitis

5 Conclusion

Our primary goal is to provide an adaptive learning support environment that will effectively accommodate a wide variety of students with different skills, background, and cognitive learning styles. The web offers dynamic and open learning environment. Based on the student-centered philosophy, personalization and adaptation are the important features for the individual Web-based learning environment and system.

In this paper a framework of individual web-based learning system is presented by focusing on learner's cognitive learning process, learning pattern and activities, as well as the technology support needed. The proposed framework utilizes genetic algorithm for representing and extracting dynamic learning process and learning pattern to support students' deep learning, efficient tutoring and collaboration in web-based learning environment. We adopt the genetic algorithm to extract learning pattern from the system. We also compared these learning patterns extracted by this method with the learning patterns from original neural network and C4.5 decision tree. From Table 1, the mode of neural network has higher precision, because the training set offered to neural network and C4.5 decision tree has noise, whereas the algorithm in this paper has the best ability of generalization and noise tolerance. In addition, we present the structural model of web-based learning system by applying Pervasive Agent Ontology, and from which we adopt OWL language to describe domain ontology knowledge and uses Jena inference engine to infer and extract answers. And experiments do prove that it is feasible to use the method to develop an individual Web-based learning system, which is valuable for further study in more depth.

Acknowledgments. We would like to acknowledge the support from the Foundation of Post-Doctor in China, the National Natural Science Foundation of China (90412010, 60573166), the National High Technology Research and Development

Program (863 Program in china: 2004AA1Z2450), HP Labs China under “On line course organization”, Special Fund supported by the Doctoral Program of Higher Education of China for Fast Sharing of Science Paper in Net Era by CSTD under Grant No.20070001073.

References

1. Mirabella, V., Kimani, S., Catarci, T.: A No Frills Approach for Accessible Web Based Learning Material. In: Proc. of the 13th International World Wide Web Conference, New York, pp. 89–95 (2006)
2. Tim, B., James, H.: The Semantic Web. Scientific American (2006)
3. Holsapple, C.W., Joshi, K.D.: A Collaborative Approach to Ontology Design. *Int. J. Communications of the ACM* 2, 42–47 (2007)
4. Brusilovsky, P.: KnowledgeTree: A Distributed Architecture for Adaptive E-learning. In: Proc. of WWW 2004, pp. 104–111. ACM, New York (2004)
5. Brusilovsky, P., Anderson, J.: An Adaptive System for Learning Cognitive Psychology on the Web. In: Proceedings WebNet 2007- 7th World Conference of the WWW, Internet & Intranet, Orlando, Florida, pp. 92–97 (2007)
6. Lisa, F.: A Cognitive Approach of Web-based Learning Support Systems. In: Proceedings ICALT 2005-5th IEEE International Conference on Advanced Learning Technologies, August 2005, pp. 233–239. IEEE computer society, Los Alamitos (2005)
7. Zhongzhi, S.: Knowledge Discovering, 1st edn. Qinghua University Press, Beijing (2002)
8. Corcoran, A., Sen, S.: Using Real-valued Genetic Algorithms to Evolve Rule Sets for Classification. In: Proceeding of the eleventh IEEE International Conference on Evolutionary Computation 11, New York, USA, pp. 650–661 (2006)
9. Brown, P., Della, P.: Class-based N-gram Models of Natural Language. *International Journal of Computational Linguistics* 28(4), 477–480 (2006)

Large Population Size IGAs with Individuals' Fitness Not Assigned by User

Jie Yuan* and Dunwei Gong

School of Information and Electrical Engineering,
China University of Mining and Technology, 221008, Xuzhou, China
sweet_joy@163.com, dwgong@vip.163.com

Abstract. User fatigue problem in traditional interactive genetic algorithms restricts its population size. It is necessary to maintain a large population size to optimize complicated problems. We present a large population size interactive genetic algorithm with individuals' fitness not assigned by user in this paper. The algorithm divides the population into several clusters, and the maximum number of clusters is changeable with the evolution and distribution of the population. The user only evaluates a center individual in each cluster and others' fitness is estimated based on these ones. In addition, to assign a center individual's fitness, we record the time when the user evaluates it satisfactory or unsatisfactory according to his/her sensitiveness, and its fitness is automatically calculated based on the time. Finally, we apply the proposed algorithm to the one-max problem, and compare it with traditional interactive genetic algorithms. The experimental results validate its efficiency.

Keywords: Optimization; genetic algorithm; interactive genetic algorithm; cluster; user fatigue.

1 Introduction

Interactive genetic algorithms (IGAs), proposed in mid 1980s, are effective methods to solve optimization problems with implicit or fuzzy indices [1]. Involving human's intelligence the fitness of individuals is assigned by human rather than a function which is difficult or even impossible to express explicitly. IGAs have applied successfully in many fields, such as face indication [2], fashion design [3], music composition [4], hearing aid fitting [5], and so on.

IGAs, different from GAs requiring hundreds of generations and a large population size to obtain the expectant performance, often have small population size and few evolutionary generations [6]. Thus, to improve performance of IGAs could start with breaking the limitation of population size. In fact some researchers have employed large population to evolve [7][8]. They classified large population size into several clusters based on K-means clustering, and user evaluated the centers of these clusters. Other individuals were estimated based on

* This work was supported by NSFC grant No.60775044 and Program for New Century Excellent Talents in University.

their similarities to their centers. But it is difficult to determine an appropriate value of the key parameter K in advance. In addition these approaches only use a cluster's center to evaluate individuals in the same cluster without employing the information of the individuals belonging to different clusters. Therefore, we present a clustering method that makes full use of information of the population and the number of clusters is changeable with the distribution of a population. Except the center individuals evaluated by user, the rests are estimated by the system automatically fully utilizing the information of center individuals.

There are many approaches to alleviate user fatigue in IGAs, but few researchers adopt an appropriate value to express an individual's fitness. Based on uncertain and fuzzy cognition of human on an individual, Gong *et al.* adopted interval numbers to express fitness [9], lightening a burden resulting from evaluating an individual. While, the user still has to give two values of endpoints to each individual. An eye-tracking device was applied to measure human preference and to obtain individuals' fitness [10]. It can effectively alleviate user fatigue, but an eye-tracker is still expensive and hardly popularized. In this paper, we adopt a novel evaluation method that user only evaluates an individual satisfactory or unsatisfactory. Considering a user's different sensitiveness to the beautiful and the ugly, we can record the time spent by the user in evaluating an individual, and the fitness is calculated automatically according to the user's evaluation and the time. Without assigning the fitness directly by user and, adopting the clustering method can alleviate user fatigue and improve the performance of IGAs. The following section presents the proposed algorithm in detail.

2 Large Population Size IGAs with Individuals' Fitness Not Assigned by User (LPS-IGAs-IFNAU)

2.1 Ideas of Large Population Size in IGAs

Denote a population in the t -th generation as $x(t)$ with its population size N_i . Assume that $x(t)$ can be divided into $N_c(t)$ clusters and $N_c(t) \leq N_{\max c}(t)$, where $N_{\max c}(t)$ is the allowable maximum number of clusters in the t -th generation. The genotype of an individual $x_i(t)$ can be expressed as $x_{i1}x_{i2} \dots x_{iN_g}$, where $x_{im}(m = 1, 2, \dots, N_g)$ is a gene meaning unit [11]. Using a distance measure based on gene meaning units a large population is divided into several clusters. The similarity degree of $x_i(t)$ and $x_j(t)$ is calculated as follows.

$$\alpha(x_i(t), x_j(t)) = \frac{1}{N_g} \sum_{m=1}^{N_g} \alpha_m(x_i(t), x_j(t)) \quad (1)$$

where

$$\alpha_m(x_i(t), x_j(t)) = \begin{cases} 1 & \text{if } x_{im} = x_{jm} \\ 0 & \text{others} \end{cases} .$$

If $\alpha(x_i(t), x_j(t)) \geq \alpha_0$, we consider that $x_i(t)$ and $x_j(t)$ belong to the same cluster, where $\alpha_0 \in (0, 1]$ is a threshold reflecting the degree of similarity.

The following shows the process of clustering. Firstly randomly choose an individual $x_i(t)$, then looking for all individuals $x_j(t)(j = 1, 2, \dots, N_i, j \neq i)$ satisfying $\alpha(x_i(t), x_j(t)) \geq \alpha_0$, and obtain the first cluster denoted as $\{c_1(t)\}$, namely $\{c_1(t)\} = \{x_j(t) | \alpha(x_i(t), x_j(t)) \geq \alpha_0, x_j(t) \in x(t)\}$. If denote $c_1(t)$ as $\{c_1(t)\}$'s center, we have $c_1(t) = x_i(t)$. After that let $x(t) \leftarrow x(t) \setminus \{c_1(t)\}$, and randomly choose another individual, perform the same process as the above to obtain the second cluster $\{c_2(t)\}, \dots$, until $x(t)$ has no individual or $N_c(t) = N_{\maxc}(t)$. During clustering, when $N_c(t) = N_{\maxc}(t) - 1$ and $x(t)$ still has individuals not being clustered, we will put all the rest into the N_{\maxc} -th cluster.

After clustering $x(t)$ is divided into $\{c_1(t)\}, \{c_2(t)\}, \dots, \{c_{N_c(t)}(t)\}$, and respectively their centers are $c_1(t), c_2(t), \dots, c_{N_c(t)}(t)$, which are evaluated by user. N_{\maxc} as an important parameter in clustering should be set reasonably. We expect that N_{\maxc} is related to the distribution of individuals in $x(t)$; the more even the individuals' distribution is, the larger N_{\maxc} is. What's more, in the later phase of evolution we expect good convergence of a population, which requires fewer clusters, namely the smaller N_{\maxc} . In order to reflect its features, firstly denote a function $SD(t)$ as the similarity degree of individuals in $x(t)$.

$$SD(t) = \frac{2}{N_i(N_i - 1)} \sum_{i=1}^{N_i-1} \sum_{j=i+1}^{N_i} \alpha(x_i(t), x_j(t)) . \tag{2}$$

It is easy to obtain that $SD(t) \in [0, 1]$. The more similar the individuals are in $x(t)$, the closer $SD(t)$ approaches to 1, and $x(t)$ should be divided into fewer clusters. The allowable maximum number of clusters in the t -th generation is computed as follows.

$$N_{\maxc}(t) = \lceil N_{\maxd} - t \cdot SD(t) \cdot e^{-\frac{t}{T_{\max}}} \rceil . \tag{3}$$

Where N_{\maxd} is the largest number of individuals displayed through human-computer interface, which is often the population size of traditional interactive genetic algorithms (TIGAs), T_{\max} is the allowable maximum evolutionary generations and, $\lceil \cdot \rceil$ is the upper integer function. In order to make sure N_{\maxc} be positive, $t \cdot SD(t) \cdot e^{-\frac{t}{T_{\max}}} < N_{\maxd}$ should be met. And we can get

$$t \cdot SD(t) \cdot e^{-\frac{t}{T_{\max}}} \leq T_{\max} \cdot SD(T_{\max}) \cdot e^{-1} \leq T_{\max} \cdot e^{-1} .$$

Therefore to meet the constraint condition, the inequality $T_{\max} \cdot e^{-1} < N_{\maxd}$ should be met, namely $T_{\max} < e \cdot N_{\maxd}$.

2.2 Strategies to Obtain the Fitness of Clusters' Center Individuals

A simple approach to evaluation is adopted; the user only gives either a satisfactory or unsatisfactory reaction to each individual presented in a certain sequence based on his/her sensibility. For an individual, the more the user prefers it, the less time he/she spends to evaluate it, thus the greater its fitness should be. Similarly, the more the user dislikes an individual, the less time spent by user

choosing, hence the smaller its fitness should be. Based on these, we can obtain an individual's fitness through a map from time space to fitness space.

Let $T(x(t))$ be the time when the evolutionary system displays a population to user and, $S_s(t)$ and $S_u(t)$ are satisfactory set and unsatisfactory set respectively. Individuals discussed in this subsection are center individuals $(c_i(t), i = 1, 2, \dots, N_c(t))$. Denote the time when $c_i(t)$ is stored in $S_s(t)$ or $S_u(t)$ as $T(c_i(t))$. Thus human spends $T(c_i(t)) - T(x(t))$ in evaluating $c_i(t)$. In order to calculate the fitness, denote a scalar $g(c_i(t))$. For $c_i(t)$ in $S_s(t)$,

$$g(c_i(t)) = \max_{c_k(t) \in S_s(t)} T(c_k(t)) - T(c_i(t)) + \max_{c_j(t) \in S_u(t)} T(c_j(t)) - \min_{c_j(t) \in S_u(t)} T(c_j(t)) . \tag{4}$$

And for $c_i(t)$ in $S_u(t)$,

$$g(c_i(t)) = T(c_i(t)) - \min_{c_j(t) \in S_u(t)} T(c_j(t)) . \tag{5}$$

If the fitness of individuals is confined to the range of $[f_{\min}, f_{\max}]$, the fitness of $c_i(t)$ is calculated as follows:

$$f(c_i(t)) = f_{\min} + (f_{\max} - f_{\min}) \cdot \frac{g(c_i(t))}{h(t)} . \tag{6}$$

where

$$h(t) = \max_{c_k(t) \in S_s(t)} T(c_k(t)) - \min_{c_k(t) \in S_s(t)} T(c_k(t)) + \max_{c_j(t) \in S_u(t)} T(c_j(t)) - \min_{c_j(t) \in S_u(t)} T(c_j(t)) . \tag{7}$$

An interest phenomenon is that $T(x(t))$ is absent in (4) to (7), that is to say, in order to calculate an individual's fitness, we only require the time when an individual is stored in the satisfactory or the unsatisfactory set. The order of evaluation is important for the method, whereas it is not difficult on condition that we obey the cognitive law. If we violate the general cognitive law and choose an individual in stochastic order, the algorithm will not work.

2.3 Strategies to Obtain the Fitness of Other Individuals

After clustering, it can be seen that the individuals in different clusters may be similar. Thus, other individuals' fitness could be estimated through the similar center individuals.

Assume an individual $x_j(t)$ and its estimated fitness is $\widehat{f}(x_j(t))$. Among all center individuals searching for center individuals $c_i(t)$ satisfying $\alpha(c_i(t), x_j(t)) \geq \alpha_0$, where $i = 1, 2, \dots, N_c(t)$. Denote nc_j as the number of $c_i(t)$ satisfying the inequality. And $\widehat{f}(x_j(t))$ can be estimated as follows.

$$\widehat{f}(x_j(t)) = \frac{1}{nc_j} \sum_i f(c_i(t)), \{c_i(t) | \alpha(c_i(t), x_j(t)) \geq \alpha_0\} . \tag{8}$$

2.4 Genetic Operations

Selection Operator. We adopt tournament selection for its advantages [12] with tournament size of two. Two individuals are chosen at random from the population, and the better one is chosen based on fitness.

Crossover Scheme. In order to generate high quality offspring, the two individuals to perform crossover operation should not be similar. Randomly choose two individuals, e.g. $x_m(t)$ and $x_n(t)$, and check out them to make sure they meet $\alpha(x_m(t), x_n(t)) < \alpha_0$. In addition the crossover point is randomly chosen following the principle: the exchanged segments of the two parents are not identical. We will choose another crossover point until it meets the principle.

3 Applications in the One-Max Optimization Problem

3.1 Description of the One-Max Problem

In order to verify its efficiency, the proposed algorithm is applied in the one-max problem which is borrowed from [13]. The one-max optimization problem is a simple problem consisting in maximizing the number of 1s of a bit-string. The goal of the one-max problem is to converge to the white color with its RGB values being (255,255,255). In this paper we only use brightness (M_1) in [13] as the similarity criteria: $M_1(R, G, B) = R + G + B$.

3.2 Parameter Settings

We have developed the color evolutionary optimization system based on LPS-IGAs-IFNAU by Visual Basic 6.0. The interface is as shown in Fig. 1. The individuals shown to the user are RGB colors, and each individuals genotype is encoded as a binary string of 24 bits, with 8 bits each for red, green and blue, and each 8 bits binary string indicate a decimal number within [0, 255]. In order to apply the proposed method of subsection 2.1, we have modified the similarity measure function which is described as follows. For two individuals $x_i(t)$ and $x_j(t)$, their RGB values are (R_i, G_i, B_i) and (R_j, G_j, B_j) respectively.

$$\alpha(x_i(t), x_j(t)) = \frac{255 \times 3 - (|R_i - R_j| + |G_i - G_j| + |B_i - B_j|)}{255 \times 3} . \quad (9)$$

The goal of the experiment is to find the white color with RGB values being (255,255,255), but as we know, the ability of users to distinguish different colors is not same [13]. Therefore we ease the restriction of white, namely if $M_1 \geq 245 + 245 + 245 = 735$, we consider that the white color is found. Table 1 shows the parameter settings. T_{\max} is the other terminative criterion. As deduced in subsection 2.1, T_{\max} should satisfy $T_{\max} < 2.72 \times 12 \approx 32$.

In order to compare the performance of LPS-IGAs-IFNAU and TIGAs, another color evolutionary optimization system based on TIGAs has been developed where individuals' fitness is directly assigned by the user. In TIGAs

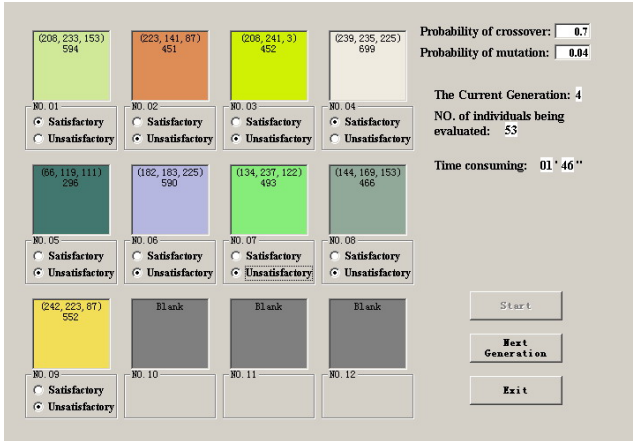


Fig. 1. Screenshot of interactive one-max problem based on LPS-IGAs-IFNAU

$N_i = N_{\max d} = 12$. The same parameter settings as Table 1 presented and the elitist strategy are used in the system. For two individuals with the same genotype, if the user marks the one, the other’s fitness, equaling to the first one’s, will be automatically obtained by the program. Five volunteers with no experience on evolutionary methods and without understanding the mechanism behind evolutionary programs perform these experiments, and for each algorithm each person conducts 5 independent runs.

Table 1. Parameter settings

Parameter	Value	Parameter	Value
N_i	200	f_{\min}	0
$N_{\max d}$	12	f_{\max}	100
p_c	0.7	α_0	0.87
p_m	0.04	T_{\max}	20

3.3 Results

Table 2 presents the experimental results of all 25 runs in summarized form. The last row shows the advantages of LPS-IGAs-IFNAU. It can be seen from Table 2 that both the algorithms can achieve the goal, but LPS-IGAs-IFNAU has over twice the success rate than TIGAs. Time-consuming of LPS-IGAs-IFNAU is approximate 3 times faster. Also employing the large population size does not increase the number of individuals being evaluated by user.

Table 3 presents the comparison of the effective runs which indicate the runs where the algorithms can achieve the goal. Having been shown as Table 2, there are 19 effective runs in LPS-IGAs-IFNAU and there are only 9 runs in TIGAs. The number of generations required by LPS-IGAs-IFNAU is a little more than

Table 2. Performance comparison of all 25 runs

Algorithm	Genera- tions	Time-consuming (m's")	NO. of individuals be- ing evaluated by user	Success rate to find the goal
LPS-IGAs-IFNAU	14	2'52"	117	76% (19/25)
TIGAs	16	7'48"	198	36% (9/25)
Speedup	1.14	2.72	1.69	2.11

Table 3. Performance comparison of effective runs

Algorithm	Genera- tions	Time to converge (m's")	NO. of individ- uals being eval- uated by user	Time-consuming per generation (s)	Time-consuming per individual (s)
LPS-IGAs- IFNAU	13	2'44"	107	12.6	1.53
TIGAs	10	4'31"	117	27.1	2.32
Speedup	0.77	1.65	1.09	2.15	1.52

that by TIGAs, but the convergent time of LPS-IGAs-IFNAU is shorter, because of its small time-consuming per generation. In addition, the time-consuming per individual by LPS-IGAs-IFNAU also shows its advantages.

Employing large population size without increasing user's burden of evaluation, the number of individuals being searched is much larger ($N_i/N_{\max d} \approx 17$) than that in TIGAs. And it drives the algorithm to have more opportunities to find the satisfactory individuals than TIGAs. Considering the evaluation mode by user, the traditional method is not conform to the user's cognition particularly resulting in user fatigue. For the fuzzy of user's cognition, it is much easy to make an alternative choice for the user. And the results show that our algorithm is able to effectively alleviate the user fatigue and has better performance, such as convergence and rapidity, than TIGAs. The consensus of five volunteers is that the assignment method and the strategy of changeable number of colors in LPS-IGAs-IFNAU are very useful. Although more generations are needed, the evaluation process is very simple, and during the evaluation the users are relaxed without mental burden.

4 Conclusions

The aim of the proposed algorithm is to alleviate user fatigue and to improve IGA's performance, and the experimental results show its advantages.

The number of clusters decreases during the evolution. Especially at the latter stage of the evolution resulting from the elitist strategy, some individual always

maintains the best but is not the desired one. Other individuals belonging to its cluster have few opportunity to be presented to the user, among which there may be the desired one. Therefore it is reasonable to divide the evolutionary process into several stages, and the population in different stages has different population size. What's more, resulting from the clustering method, individuals in the $N_{\max}(t)$ -th cluster may not be similar to the center individual, and how to deal with the cluster more reasonably is our future research.

References

1. Dawkins, R.: *The Blind Watchmaker*. Longman, U.K (1986)
2. Caldwell, C., Johnston, V.S.: Tracking a Criminal Suspect through 'Face-space' with a Genetic Algorithm. In: 4th International Conference on Genetic Algorithms, pp. 416–421. Morgan Kaufmann Press, San Francisco (1991)
3. Kim, H.S., Cho, S.B.: Application of Interactive Genetic Algorithm to Fashion Design. *Eng. Appl. Artif. Intell.* 13, 635–644 (2000)
4. Tokui, N., Iba, H.: Music Composition with Interactive Evolutionary Computation. In: 3rd International Conference on Generative Art, pp. 215–226 (2000)
5. Takagi, H., Ohsaki, M.: Interactive Evolutionary Computation-based Hearing Aid Fitting. *IEEE Trans. Evol. Comput.* 11, 414–427 (2007)
6. Takagi, H.: Interactive Evolutionary Computation: Fusion of the Capabilities of EC Optimization and Human Evaluation. In: *IEEE*, pp. 1275–1296. IEEE Press, New York (2001)
7. Lee, J.Y., Cho, S.B.: Sparse Fitness Evaluation for Reducing User Burden in Interactive Genetic Algorithm. In: *IEEE International Fuzzy Systems Conference*, pp. 998–1003. IEEE Press, New York (1999)
8. Kim, H.S., Cho, S.B.: An Efficient Genetic Algorithm with Less Fitness Evaluation by Clustering. In: *IEEE Congress on Evolutionary Computation*, pp. 887–894. IEEE Press, New York (2001)
9. Gong, D.W., Guo, G.S.: Interactive Genetic Algorithms with Interval Fitness of Evolutionary Individuals. *Dynamics of Continuous, Discrete and Impulsive Systems, Series B: Complex Systems and Applications-modeling, Control and Simulations* 14(s2), 446–450 (2007)
10. Pallez, D., Collard, P., Baccino, T., Dumercy, L.: Eye-tracking Evolutionary Algorithm to Minimize User Fatigue in IEC Applied to Interactive One-max Problem. In: *Genetic and Evolutionary Computation Conference*, pp. 2883–2886. ACM, New York (2007)
11. Gong, D.W., Hao, G.S., Zhou, Y., Guo, Y.N.: *Theory and Applications of Interactive Genetic Algorithms (in Chinese)*. Defense Industry, Beijing (2007)
12. Mitchell, M.: *An Introduction to Genetic Algorithms*. MIT, Cambridge (1996)
13. Cheng, C.D., Kosorukoff, A.: Interactive One-max Problem Allows to Compare the Performance of Interactive and Human-based Genetic Algorithms. In: Deb, K., et al. (eds.) *GECCO 2004*. LNCS, vol. 3102, pp. 983–993. Springer, Heidelberg (2004)

Missing Data Imputation in Time Series by Evolutionary Algorithms

Juan C. Figueroa García¹, Dusko Kalenatic², and Cesar Amilcar Lopez Bello³

¹ Universidad Distrital Francisco José de Caldas, Bogotá - Colombia
jcfgueroag@udistrital.edu.co

² Universidad de la Sabana, Chia - Colombia
Universidad Católica de Colombia, Bogotá - Colombia
duskokalenatic@yahoo.com

³ Universidad Distrital Francisco José de Caldas, Bogotá - Colombia
Universidad de la Sabana, Chia - Colombia
clopezb@udistrital.edu.co

Abstract. This paper presents a proposal based in an Evolutionary algorithm for imputing missing observations in Time Series. A genetic algorithm based on the minimization of an error function derived from their autocorrelation function, mean and variance, is presented.

All methodological aspects of the genetic structure are presented. An extended explanation of the design of the Fitness Function is provided. Four application examples are provided and solved by the proposed method.

1 Introduction and Motivation

During the last 50 years, both financial and biological Time Series have been recognized as complex prediction cases because they may contain missing observations. This data loss is an important problem for univariate time series analysis since most of available estimation methods does not provide suitable results.

An evolutionary structure is proposed for imputing missing observations in time series since it is an efficient computational intelligence tool that provides a fast exploration of the search space. To do so, a multi-criteria fitness function extracted from the autocorrelation, mean and variance of the series is minimized.

The scope of this work is to impute missing observations to an incomplete time series by using an evolutive structure without generality loss. It means that original properties of available data are not changed when the estimated data is imputed on. Four application examples are solved by using this proposal.

2 Autocovariance and Autocorrelation Functions

In a time series context, the most important order statistics to construct optimal models as ARIMA, ARCH and GARCH are the Mean, Variance and Autocorrelation functions of the process. First, some useful definitions are given below.

Definition 1. A Time Series is a set of observations x_t , each one being recorded at a specific time t . A time series model for the observed data $\{x_t\}$ is a specification of joint distribution (or possibly the means and covariances) of a sequence of random variables $\{X_t\}$ for which $\{x_t\}$ is postulated to be a realization.

The Sample Autocovariance and Sample Autocorrelation of the series are useful statistics to estimate most of the linear models as ARIMA, ARCH and GARCH. They are linear distances between the measured variable at a specific time $\{x_t\}$ to itself at a lag h , $\{x_{t+h}\}$.

Definition 2. The sample Autocovariance function $\hat{\gamma}(h)$ is:

$$\hat{\gamma}(h) = \sum_{t=1}^{n-|h|} \frac{(x_{t+|h|} - \bar{x})(x_t - \bar{x})}{n}, \quad -n < h < n \tag{1}$$

Definition 3. The sample Autocorrelation function $\hat{\rho}(h)$ is:

$$\hat{\rho}(h) = \frac{\hat{\gamma}(h)}{\hat{\gamma}(0)}, \quad -n < h < n \tag{2}$$

For further information see Graybill & Mood in [1], Wilks in [2], Huber in [3], Grimmet in [4], Ross in [5] and Brockwell & Davis in [6].

As always, these definitions are based on a strong supposition about the series: *The time series does not contain missing observations.* When one or more observations are lost, they are not *sufficient statistics*, causing bias of the real properties of the series and misrepresentation problems.

3 Genetic Structure

The selected strategy to find missing data on a multi-criterion context is an evolutionary algorithm based on a genetic structure. There are a vast amount of bibliography in evolutionary optimization, but this work is focused in designing a suitable fitness function based on available data and their statistical properties.

3.1 Fitness Function Operator

Suppose that in a time series vector $\{x_t\}$ m missing observations exist located by an index vector v , where $1 \leq v \leq n$. A vector of estimates of these missing observations is called $\{y_t\}$, where $y_t = 0$ when $t \notin v$ and $y_t = x_j$ when $t \in v$, x_j is the j th element of y_t , $1 \leq j \leq m$ located in the v_{th} position. By using these two vectors it is possible to complete the series namely $\{\hat{x}_t\}$:

$$x_t + y_t = \hat{x}_t \tag{3}$$

Now, the principal issue is to find a vector y_t which does not change the available data properties. To that effect, Autocorrelation Function, Mean and Variance of the available data are used for constructing the fitness function.

Remark 1. $\hat{\gamma}(h)$ is not calculable if there are missing data, so the selected strategy is to use a subset of x_t which is their largest and recent subset, that is:

$$\hat{\gamma}(h)^l = \sum_{t=n_1}^{n_2-|h|} \frac{(x_{t+|h|} - \bar{x})(x_t - \bar{x})}{n}, \quad (n_1 + n_2) < h < (n_2 - n_1), n_1, n_2 \in T \quad (4)$$

Where n_1 and n_2 are the lower and upper bounds of the index t and $\hat{\gamma}(h)^l$ is the autocorrelation of the largest and recent complete subset of x_t , denoted by l .

Remark 2. The mean and variance used to construct the fitness function are obtained by removing the missing data from the original series, and then a vector $\{x_i\}$ of size $n - m \times 1$ is defined as follows.

$$\bar{x}^a = \sum_{i=1}^{n-m} \frac{x_i}{n - m} \quad (5)$$

$$Var(x^a) = \sum_{i=1}^{n-m} \frac{(x_i - \bar{x}^a)^2}{n - m - 1} \quad (6)$$

Where \bar{x}^a and $Var(x^a)$ are the mean and variance of the available dataset.

Thus, the main goal is to find an evolutionary estimate of the m missing observations that does not change its $\hat{\gamma}(h)^l$, \bar{x}^a and $Var(x^a)$ statistics. Now the fitness function \mathcal{F} can be expressed as:

$$\mathcal{F} = \sum_{h=1}^H [| \hat{\rho}(h)^l - \hat{\rho}(h) |] + | \bar{x}^a - \bar{x} | + | Var(x^a) - Var(x) | \quad (7)$$

Therefore, the principal objective of the genetic structure is to minimize \mathcal{F} .

3.2 Individuals, Population Size and Number of Generations

An *Individual* is defined as a *chromosome* of size m , where each *gen* or cell is a missing observation indexed in the v_{th} position of y_t .

The population size is defined by two values: The m missing values and a pre-selected k number of individuals, creating a matrix called $P_{k,m}^g$ where g is the *Generation index*. Usually, the size of k impacts their efficiency directly.

The generation index g is a genetic operator which is used as stop criterion. In this paper, the maximum number of generations is defined as G .

3.3 Population Random Generator

The selected Random Generator is called R_j . The original *pdf* of the series without missing data $\{x_i\}$ can be used as a random number generator, but in practice, the uniform generator which is presented next, exhibits better results:

$$R_j(a, b) = a + r_j(b - a)I_{[0,1]}(r_j) \quad (8)$$

Law and Kelton (See [7]) do an important discussion about pseudo-randomly number generation showing the uniform generator as a proper method to simulate variables. Therefore it is possible to use it instead of the sample distribution.

3.4 Mutation and Crossover Operators

- The selected Mutation strategy is described below:
 1. Select a random position for each orderly individual in $P_{k,m}^g$ by \mathcal{F} .
 2. Replace the selected position with a new individual by using (8).
 3. Repeat Step 3 for the c_1 better individuals of each population $P_{k,m}^g$.
- The selected Crossover strategy is described below:
 1. Select the c_2 first individuals in the orderly Population $P_{k,m}^g$ by \mathcal{F} .
 2. Generate a new individual by replacing all even genes with its respective even gene located in the next individual.
 3. Generate a new individual by replacing all odd genes with its respective odd gene located in the next individual for each one.
 4. Repeat Step 3 for the c_2 better individuals of each population $P_{k,m}^g$.

To complete the population, an additional set of individuals is generated by replacing worst individuals with new individuals, trying an exhaustive search. First, the best four individuals are preserved for the next generation and later a complementary population of size $\{k - 4 - c_1 - c_2 \times m\}$ is obtained by using (8).

3.5 Finalization and Stopping Strategy

Two stopping criterions are used to stop the algorithm: A first one is by defining a maximum number of iterations called G , that is $g \rightarrow G$, and the second one stops when \mathcal{F} has no significant improvement through a specific number of iterations.

Finally, the best individual is selected by \mathcal{F} and is imputed in the original series, obtaining a complete dataset.

4 Implementation in Some Study Cases

Four weather time series with missing observations are imputed by using the EM algorithm and the proposed Genetic approach. The available four datasets are displayed in Gray Dashed Line in the Figure 1.

Here, a) is the daily variation for the Mean Temperature (MT), b) is the Solar Brightness (SB), c) is the Mean Relative Humidity (MRH) and d) is the Min-Absolute Relative Humidity ($MARH$) in the town of Chia - Colombia. All missing observations are caused by problems on devices and other human factors.

4.1 Statistical Approaches

Most of the statistical methods to impute missing data are based on error functions derived from mean, variance or Likelihood Ratios of the series. However, the time series case is more complex due to their autocorrelation structure.

First, some statistics obtained from the available data are shown in the Table 1. \mathcal{F} is obtained by using (5) and (6) for each genetic structure

This Table shows all required statistics for constructing \mathcal{F} . The autocorrelation structures, means and variances of all series are shown.

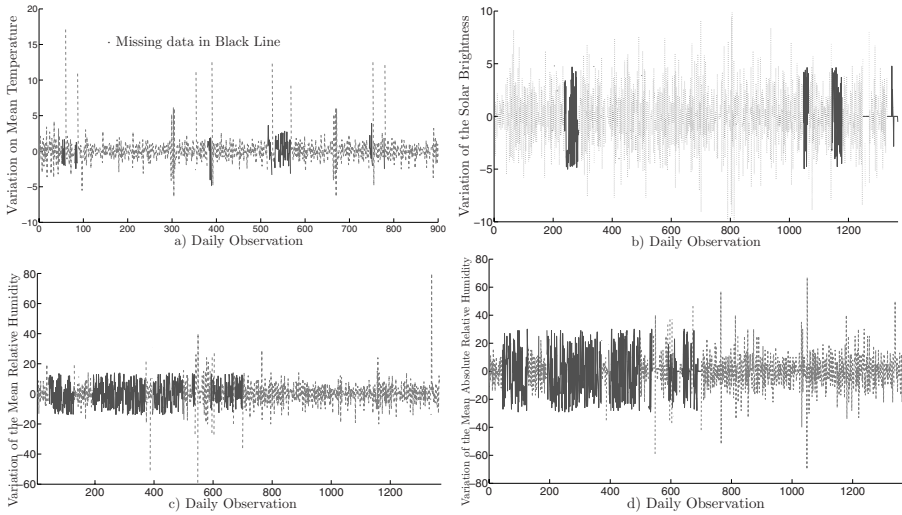


Fig. 1. Complete Data Set with Imputed Missing Data

Table 1. Observed statistic

Series	$\hat{\rho}(1)^t$	$\hat{\rho}(2)^t$	$\hat{\rho}(3)^t$	$\hat{\rho}(4)^t$	$\hat{\rho}(5)^t$	$\hat{\rho}(6)^t$	\bar{x}^a	$Var(x^a)$	m	n	N
MT	-0.382	-0.036	0.015	0.013	-0.072	0.023	0.096	3.115	81	819	900
SB	-0.39	0.036	-0.015	-0.15	0.1053	0.2106	0.0022	7.3432	163	1204	1367
MRH	-0.062	-0.251	-0.049	0.062	0.0029	-0.035	0.198	62.107	480	887	1367
MARH	0.6388	0.5056	0.4031	0.3771	0.3248	0.3517	0.1478	138.41	481	886	1367

4.2 Missing Data Estimates and the EM Algorithm

Most of the optimal imputation algorithms as the EM^{\square} algorithm are based on conditional expectations of a random variable, obtained from a set of auxiliary variables which get an estimate of the missing data (See Dempster in [8]). Their principal objective is to maximize a *Likelihood Function* of the *pdf* sample.

Remark 3. Note that the EM algorithm should have at least one auxiliary variable to compute the conditional expectations. If this vector is not available, it will replace all missing values with $S(\theta, \theta_{k-1}) = E_{\theta_{k-1}}\{\log f(X; \theta) | X = y\} = E_{\theta_{k-1}}(Y)$. It is a strong problem since a single estimate replacing all missing observations cause variations on their autocorrelation function.

Gaetana & Yao in [9] propose a variation of the EM algorithm based in a simulated annealing approach to improve its efficiency for the multivariate case. Celeux & Diebolt in [10], Levine & Casella in [11], Nielsen in [12] have reported some modifications for a stochastic scenario and Arnold in [13] estimates the parameters of a state-dependant AR model by using the EM algorithm.

¹ Acronym used for the *Expectation Maximization* Algorithm.

Table 2. EM Estimation

$\hat{\rho}(2)$	$\hat{\rho}(3)$	$\hat{\rho}(4)$	$\hat{\rho}(5)$	$\hat{\rho}(6)$	\bar{x}	$Var(x)$	EM It.	\mathcal{F}
MT	-0.187	-0.034	-0.05	0.067	-0.074	0.082	0.096	2.834 4 0.897
SB	-0.414	0.00014	-0.0884	0.0616	-0.0577	0.0382	0.00198	6.467 4 1.633
MRH	-0.1923	-0.0396	-0.0947	-0.038	0.019	-0.0008	0.1988	40.19 3 22.45
MARH	0.6404	0.5186	0.4334	0.381	0.3193	0.3262	0.1507	156.2 5 17.88

In this way, the EM algorithm replace all missing observations with $E(X) = \bar{x}_t$. Their statistical properties are presented in the Table 2.

This Table shows great differences among obtained variances, $\hat{\rho}(1)$ and their available values which result on high values of \mathcal{F} . So it is clear that EM algorithm is not a proper way to impute missing data on an Univariate Time Series context.

4.3 Genetic Approach

Few applications of genetic algorithms to impute missing data have been reported. Abdella & Marwala in [14] use a genetic structure in databases, Eklund in [15] solves a spatial data problem in a Gas Turbine by using a genetic-hill-climbing hybrid structure. Siripitayananon, Hui-Chuan & Jin Kang-Ren in [16] use Neural Networks and Parveen & Green in [17] use Recurrent Networks.

Now, the algorithm proposed in above Section is implemented. The Table 3 shows the crossover, mutation and remaining parameters for each case.

The obtained solution is displayed in black in the Figure 1 and their properties are presented in the Table 4.

Three sizes $k = \{100, 500, 1000\}$ per population are implemented. By using $k = 100$, both computing time and quality of solutions are improved. These results allow to view the evolutionary strategy as an appropriate way to estimate missing data since it does not change their original properties.

Table 3. Evolutionary Algorithm Parameters

Series	k	a	b	c_1	c_2	n_1	n_2	m	G	Time (sec)
MT	100	-5	5	4	4	780	900	81	1000	1000
SB	100	-5	5	4	4	1270	1331	163	5000	5000
MRH	100	-10	10	4	4	1341	1367	480	3000	3000
MARH	100	0	75	4	4	823	1337	481	1500	1500

Table 4. Evolutionary Results

Series	$\hat{\rho}(1)$	$\hat{\rho}(2)$	$\hat{\rho}(3)$	$\hat{\rho}(4)$	$\hat{\rho}(5)$	$\hat{\rho}(6)$	\bar{x}	$Var(x)$	\mathcal{F}
MT	-0.286	-0.036	0.015	0.013	-0.072	0.023	0.093	3.115	0.1134
SB	-0.396	0.0216	-0.0258	-0.0063	0.0175	-0.00573	0.00224	7.3432	0.4665
MRH	-0.0806	-0.1248	-0.0534	-0.0002	0.00016	-0.02276	0.1981	61.107	0.2289
MARH	0.4439	0.3495	0.3027	0.2563	0.1988	0.22349	0.1476	138.74	1.1566

5 Output Analysis

Some tests on randomness, means, Variances and autocorrelations for both data groups in each series are implemented in MatLab[®] and shown in the Table 5.

Table 5. Tests on Normality and Randomness

Series	<i>S-W</i>	<i>K-S</i>	<i>Ljung-Box</i> ^b	<i>McLeod-Li</i> ^b	<i>AICC-YW</i>
<i>MT</i>	≈ 0	≈ 0	0.0001	0.9283	1
<i>SB</i>	≈ 0	≈ 0	≈ 0	≈ 0	9
<i>MRH</i>	≈ 0	≈ 0	0.241	≈ 1	2
<i>MARH</i>	≈ 0	≈ 0	≈ 0	≈ 0	11

^b This Test is made by using the first lag of the series.

5.1 Tests on Means and Variances

To verify statistical differences between both original and imputed series, some tests are applied. The hypothesis considered here are:

Table 6. Hypothesis on Means and Variances

	<i>Test on Means</i>	<i>Test on Variances</i>
$H_0 :$	$\mu = \mu^a$	$Var(x) = Var(x^a)$
$H_a :$	$\mu \neq \mu^a$	$Var(x) \neq Var(x^a)$

The obtained results of the Test on Means are presented in the Table 7.

Table 7. Tests on Means

Test.	<i>MT p-value</i>	<i>SB p-value</i>	<i>MRH p-value</i>	<i>MARH p-value</i>
Welch	0.972	≈ 1	≈ 1	0.662
Brown-Forsythe	0.972	≈ 1	≈ 1	0.662
K-S.	≈ 1	0.994	0.074	≈ 0
Mann-Whitney	0.832	0.971	0.995	0.738

With these statistical evidences, the Hypothesis on means defined in the Table 6 with an $\alpha = 0.05$ of confidence is Accepted. To contrast their variances, the ANOVA and Levene tests are implemented. Their results are shown in the Tables 8 and 9 respectively.

The ANOVA and Levene tests concludes that there are no differences between $\mu \rightarrow \mu^a$, and $Var(x) \rightarrow Var(x^a)$ respectively for each series, it means that the genetic solution has no statistical differences to available data.

With these statistical evidences, the Hypothesis on variances defined in the Table 6 with an $\alpha = 0.05$ of confidence is Accepted.

Table 8. ANOVA Test

Stat.	<i>MT</i>	<i>SB</i>	<i>MRH</i>	<i>MARH</i>
<i>f</i> stat	≈ 0	≈ 0	≈ 0	0.191
<i>p</i> -value	0.972	≈ 1	≈ 1	0.662

Table 9. Levene Test

Stat.	<i>MT</i>	<i>SB</i>	<i>MRH</i>	<i>MARH</i>
Levene Stat.	0.0037	0.001	5.736	0.017
<i>p</i> -value	0.972	0.975	0.8441	0.41

The genetic solution outperforms the solution provided by the *EM* algorithm and most of the original properties of the series are preserved. All tests on means and variances does not have any statistical evidence to reject H_0 , only the *K-S* test is rejected for the *MARH* series, but all remaining tests are accepted.

6 Concluding Remarks

After four implementations, the following concluding remarks can be suggested:

1. The proposed genetic algorithm outperforms the results of the *EM* algorithm showing better solutions without modify the properties of the series.
2. The presented evolutionary strategy is a proper way to estimate solutions for missing data in a Time Series context; its flexibility and non-linear capability turns it into a powerful tool to generate good solutions on many contexts such as multivariate data, signal or image processing problems, inclusive.
3. The proposed evolutionary algorithm gets successful results without using conditional information about series, unlike to the *EM* algorithm.
4. It is recommended to use an adequate population size to reduce the computing effort. Small sizes does not make a proper search although they achieve solutions more fast than large sizes, which do not ensure avoid the local optimum problem.
5. An important aspect is the reduction of the search space by modifying the random number generator. In this case, using a great interval $[a, b]$ of the uniform generator could start a loop start around any local optima, and a small interval could generates higher fitness values.

Finally, the reader can modify these results by constructing different fitness operators \mathcal{F} to avoid local optima, adequately to the context of the problem.

Acknowledgments. The Authors would like to thank all people who are part of the *Laboratory for Automation, Microelectronics and Computational Intelligence LAMIC* and *Mathematical Modeling Applied to Industry (MMAI)* groups of the Universidad Distrital Francisco José de Caldas in Bogotá-Colombia.

References

1. Mood, A.M., Graybill, F.A., Boes, D.C.: Introduction to the Theory of Statistics. McGraw-Hill Book Company, New York (1974)
2. Wilks, A.: Mathematical Statistics. John Wiley and Sons, New York (1962)
3. Huber, P.: Robust Statistics. John Wiley and Sons, New York (1981)
4. Grimmett, G., Stirzaker, D.: Probability and Random Processes. Oxford University Press, Oxford (2001)
5. Ross, S.M.: Stochastic Processes. John Wiley and Sons, Chichester (1996)
6. Brockwell, P., Davis, R.: Time Series: Theory and Methods. Springer, Heidelberg (1998)
7. Law, A., Kelton, D.: Simulation System and Analysis. McGraw-Hill Intl., New York (2000)
8. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum-likelihood from incomplete data via the em algorithm. *Journal of Royal Statistical Society* 39(1), 1–38 (1977)
9. Gaetan, C., Yao, J.F.: A multiple-imputation metropolis version of the em algorithm. *Biometrika* 90(3), 643–654 (2003)
10. Celeux, G., Diebolt, J.: The sem algorithm: a probabilistic teacher algorithm derived from the em algorithm for the mixture problem. *Computational Statistics Quarterly* 2(1), 73–82 (1993)
11. Levine, L.A., Casella, G.: Implementations of the monte-carlo em algorithm. *Journal of Computational Graphic Statistics* 10(1), 422–439 (2000)
12. Nielsen, S.F.: The stochastic em algorithm: Estimation and asymptotic results. *Bernoulli* 6(1), 457–489 (2000)
13. Arnold, M.: Reasoning about non-linear ar models using expectation maximization. *Journal of Forecasting* 22(6), 479–490 (2003)
14. Abdella, M., Marwala, T.: The use of genetic algorithms and neural networks to approximate missing data in database. In: *IEEE 3rd International Conference on Computational Cybernetics, 2005. ICC 2005* (2005)
15. Eklund, N.: Using genetic algorithms to estimate confidence intervals for missing spatial data. *IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews* 36(4), 519–523 (2006)
16. Siripitayananon, P., Hui-Chuan, C., Kang-Ren, J.: Estimating missing data of wind speeds using neural network. In: *IEEE (ed.) Proceedings of the 2002 IEEE Southeast Conference*, vol. 1, pp. 343–348. IEEE, Los Alamitos (2002)
17. Parveen, S., Green, P.: Speech enhancement with missing data techniques using recurrent neural networks. In: *IEEE (ed.) Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2004)*, vol. 1, pp. 733–738. IEEE, Los Alamitos (2004)

Rule-Based Analysis of Behaviour Learned by Evolutionary and Reinforcement Algorithms

Stanislav Slušný, Roman Neruda, and Petra Vidnerová

Institute of Computer Science
Academy of Sciences of the Czech Republic
Pod vodárenskou věží 2, Prague 8, Czech Republic
{slusny, roman, petra}@cs.cas.cz

Abstract. We study behavioural patterns learned by a robotic agent by means of two different control and adaptive approaches — a radial basis function neural network trained by evolutionary algorithm, and a traditional reinforcement Q-learning algorithm. In both cases, a set of rules controlling the agent is derived from the learned controllers, and these sets are compared. It is shown that both procedures lead to reasonable and compact, albeit rather different, rule sets.

Keywords: Radial Basis Function Networks; Reinforcement Learning; Evolutionary Robotics.

1 Introduction

We study intelligent behaviours that arise as a result of an agent's interaction with its environment. The ultimate goal of the process is to develop an embodied and autonomous agent with a high degree of adaptive possibilities [1]. Two main approaches to tackle this problem are currently the traditional reinforcement learning (RL) [2] and evolutionary robotics (ER) [3]. Both these approaches fall into the same category of learning algorithms that are often used for tasks where it is not possible to employ more specific supervised learning techniques. Designing an agent control mechanism is a typical example of such a problem where an instant reward of agent actions is not available. We are usually able to judge positive or negative behaviour patterns of an agent (such as finding a particular spot in a maze or hitting a wall) and evaluate it on the coarser time scale. This information is used by different learning algorithms of reinforcement type to strengthen successful partial behaviour patterns, and in the course of adaptation process, to develop an agent solving a given task.

The Q-learning approach considers discrete spaces of possible agent states and actions, and in the course of adaptation creates approximations of the optimal strategy — a way to select a particular action in a given state of an agent such that the potential (delayed) reward from the environment is maximized.

The ER approach attacks the problem through a self-organization process based on artificial evolution [4]. Control mechanisms of an agent are typically based on a neural network which provides direct mapping from agent sensors to effectors. Most of the current applications use traditional multi-layer perceptron networks [5]. In our approach we utilize local unit network architecture called radial basis function (RBF) network

which has competitive performance, more learning options, and (due to its local nature) better interpretation possibilities [67].

2 Reinforcement Learning

Let us consider an embodied agent that is interacting with the environment by its sensors and effectors. The essential assumption of RL is that the agent has to be able to sense rewards coming from the environment. Rewards evaluate taken actions, agent's task is to maximize them. There has been several algorithms suggested so far. We have used the Q-learning algorithm, which was first breakthrough of RL [8].

The next important assumption is that agent is working in discrete time steps. Symbol S will denote finite discrete set of states and symbol A set of actions. In each time step t , agent determines its actual state and chooses one action. Therefore, agent's life can be written as a sequence $o_0 a_0 r_0 o_1 a_1 r_1 \dots$ where o_t denotes observation through the sensors, $a_t \in A$ action and finally symbol $r_t \in R$ represents reward, that was received at time t . The most serious assumption of RL algorithms is the *Markov property*, which states, that agent does not need history of previous observations to make decision. The decision of the agent is based on the last observation o_t only. When this property holds, we can use theory coming from the field of *Markov decision processes* (MDP). The direct implication of Markov property is the equality of states and observations. The strategy π , which determines what action is chosen in particular state, can be defined as function $\pi : S \rightarrow A$, where $\pi(s_t) = a_t$.

Now, the task of the agent is to find optimal strategy π^* . Optimal strategy is the one, that maximalizes expected reward. In MDP, single optimal deterministic strategy always exists, no matter in what state has the agent started. The quantity $V^\pi(s_t)$ is called discounted cumulative reward. It is telling us, what reward can be expected, if the agent starts in state s_t and follows policy π : $V^\pi(s_t) = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \dots = \sum_{i=0} \gamma^i r_{t+i+1}$.

Here $0 \leq \gamma < 1$ is a constant that determines the relative value of delayed versus immediate rewards. Optimal strategy π^* can now be defined as: $\pi^* = \operatorname{argmax}_\pi \{V^\pi(s), \forall s \in S\}$. To simplify the notation, let us write $V^*(s)$ instead of symbol V^{π^*} , value function corresponding to optimal strategy π^* : $V^*(s) = \max_\pi V^\pi(s)$.

-
1. Let S be the finite set of states and A finite set of actions.
 $\forall s \in S, a \in A : Q(s, a) = 0$
 2. Process sensors and obtain state s
 3. Repeat:
 - Choose and carry out action a
 - Receive reward r
 - Obtain new state s'
 - $Q(s, a) \leftarrow r + \gamma \max_{a'} Q(s', a')$
 - $s \leftarrow s'$
-

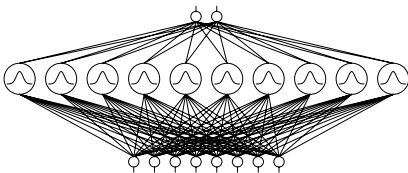
Algorithm 1. Q-learning

The Q-learning algorithm was the first algorithm to compute optimal strategy π^* [8]. The key idea of the algorithm is to define the so-called *Q-values*. $Q^\pi(s, a)$ is the expected reward, if the agent takes action a in state s and then follows policy π : $Q^\pi(s, a) = r(s, a) + \gamma V^\pi(s')$, where s' is the state, in which agent occurs taking action a in state s ($s' = \delta(s, a)$). Q-learning algorithm (Algorithm 1) guarantees convergence to optimal values of $Q^*(s, a)$, if Q-values are represented without any function approximations (in table), rewards are bounded and every state-action pair is visited infinitely often. To fulfil the last condition, every action has to be chosen with non-zero probability. Probability $P(a|s)$ of choosing action a in state s is defined as [9]: $P(a_i|s) = k^{Q(s,a_i)} / \sum_j k^{Q(s,a_j)}$, where constant $k > 0$ determines exploitation-exploration rate. Big values of k will make agent to choose actions with above average values. On the other hand, small values will make agent to choose actions randomly. Usually, learning process is started with small k , that is slightly increasing during the course of learning. Optimal values V^*s can be obtained from $Q^*(s, a)$ by the equality: $V^*(s) = \max_{a'} Q(s, a')$.

3 Evolutionary Learning of RBF Networks

Evolutionary robotics combines two AI approaches: neural networks and evolutionary algorithms. The control system of the robot is realized by a neural network, in our case an RBF network. It is difficult to train such a network by traditional supervised learning algorithms since they require instant feedback in each step, which is not the case for evolution of behaviour. Here we typically can evaluate each run of a robot as a good or bad one, but it is impossible to assess each one move as good or bad. Thus, the evolutionary algorithm represent one of the few possibilities how to train the network.

The *RBF network* [10][11][12], used in this work, is a feed-forward neural network with one hidden layer of *RBF units* and linear output layer. The network function is given in Eq. (1) (see Fig. 1). The evolutionary algorithms (EA) [4][13] represent a stochastic search technique used to find approximate solutions to optimization and search problems. They work with a population of *individuals* representing feasible solutions. Each individual is assigned a *fitness* that is a measure of how good solution it represents. The evolution starts from a population of completely random individuals and iterates in generations. In each generation, the fitness of each individual is evaluated. Individuals are



$$f_s(\mathbf{x}) = \sum_{j=1}^h w_{j_s} \varphi \left(\frac{\|\mathbf{x} - \mathbf{c}_j\|}{b_j} \right) \quad (1)$$

Fig. 1. A scheme of a Radial Basis Function Network, f_s is the output of the s -th output unit. φ is an activation function, typically Gaussian function $\varphi(s) = e^{-s^2}$.

stochastically selected from the current population (based on their fitness), and modified by means of genetic operators to form a new generation.

In case of RBF networks learning, each individual encodes one RBF network. The individual consists of h blocks: $I_{RBF} = \{B_1, \dots, B_h\}$, where h is a number of hidden units. Each of the blocks contains parameter values of one RBF units, $B_k = \{c_{k1}, \dots, c_{kn}, b_k, w_{k1}, \dots, w_{km}\}$, where n is the number of inputs, m is the number of outputs, $c_k = \{c_{k1}, \dots, c_{kn}\}$ is the k -th unit's centre, b_k the width and $w_k = \{w_{k1}, \dots, w_{km}\}$ the weights connecting k -th hidden unit with the output layer. The parameter values are encoded using direct floating-point encoding. Concerning the genetic operators, the standard *tournament selection*, *1-point crossover* and *additive mutation*¹ are used. The fitness function should reflect how good the robot is in given tasks and so it is always problem dependent. Detailed description of the fitness function is included in the experiment section.

4 Experimental Framework

In order to compare performance and properties of described algorithms, we conducted simulated experiment. Miniature robot of e-puck type [14] was trained to explore the environment and avoid walls. E-puck is a mobile robot supported by two lateral wheels that can rotate in both directions and two rigid pivots. The sensory system employs eight active IR sensors distributed around the body. Sensors return values from interval $[0, 4095]$. Effectors accept values from interval $[-1000, 1000]$. The higher the absolute value, the faster is the motor moving in either direction.

Table 1. Sensor values and their meaning

Sensor value	Meaning	Sensor value	Meaning
0-50	NOWHERE	1001-2000	NEAR
51-300	FEEL	2001-3000	VERYNEAR
301-500	VERYFAR	3001-4095	CRASHED
501-1000	FAR		

Instead of 4095 raw sensor values, learning algorithms worked with 5 preprocessed perceptions (see Tab. 1). Effector's values were processed in similar way: instead of 2000 values, learning algorithm was allowed to choose from values $[-500, -100, 200, 300, 500]$. To reduce the state space even more, we grouped pairs of sensors together and back sensors were not used at all. Agent was trained in the simulated environment of size 100×60 cm and tested in more complex environment of size 110×100 cm. We used Webots [15] simulation software.

In the first experiment, we have used Q-learning algorithm as described in Section 2. Each state was represented by a triple of perceptions. For example, the state [NEAR, NOWHERE, NOWHERE] means, that the robot sees a wall on its left side only. Action was represented by a pair [left speed, right speed].

¹ Additive mutation changes the values by adding small value randomly drawn from $(-\epsilon, \epsilon)$.

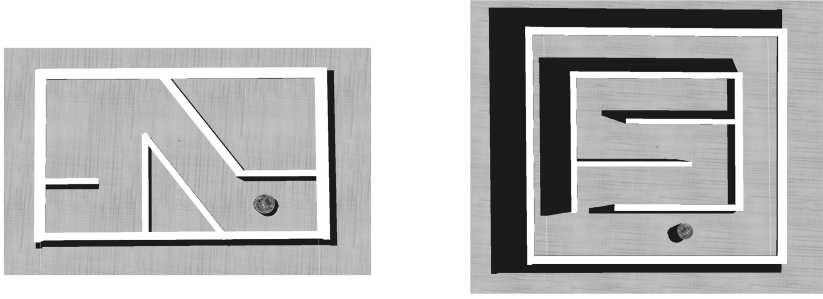


Fig. 2. Simulated environments for agent training and testing: a) Agent was trained in the simulated environment of size 100 x 60 cm. b) Simulated testing environment of size 110 x 100 cm.

Learning process was divided into episodes. Each episode took at most 1000 simulation steps. At the end of each episode, agent was moved to one from 5 randomly chosen positions. Episode could be finished earlier, if agent hit the wall. The learning process was stopped after 10000 episodes. Parameter γ was set to 0.3.

In the second experiment the evolutionary RBF networks were applied to the same maze-exploration task (see Fig. 2). The network has 3 input units, 5 hidden Gaussian units, and 2 output units. The three inputs correspond to the coupled sensor values (two left sensors, two front sensors, two right sensors), which are preprocessed in the way described in Tab. 1. The two outputs correspond to the left and right wheel speeds and before applying to robot wheels they are rounded to one of 5 valid values.

Fitness evaluation consists of two trials, which differ by agent's starting location (the two starting positions are in the opposite ends of the maze). Agent is left to live in the environment for 800 simulation steps. In each step, a three-component score is calculated to motivate agent to learn to move and to avoid obstacles:

$$T_{k,j} = V_{k,j}(1 - \sqrt{\Delta V_{k,j}})(1 - i_{k,j}). \quad (2)$$

The first component $V_{k,j}$ is computed by summing absolute values of motor speed (scaled to $\langle -1, 1 \rangle$) in the k -th simulation step and j -th trial, generating value between 0 and 1. The second component $(1 - \sqrt{\Delta V_{k,j}})$ encourages the two wheels to rotate in the same direction. The last component $(1 - i_{k,j})$ encourage obstacle avoidance. The value $i_{k,j}$ of the most active sensor (scaled to $\langle 0, 1 \rangle$) in k -th simulation step and j -th trial provides a conservative measure of how close the robot is to an object. The closer it is to an object, the higher is the measured value in range from 0.0 to 1.0. Thus, $T_{k,j}$ is in range from 0.0 to 1.0, too. In the j -th trial, score S_j is computed by summing normalized trial gains $T_{k,j}$ in each simulation step $S_j = \frac{1}{800} \sum_{k=1}^{800} T_{k,j}$. To stimulate maze exploration, agent is rewarded, when it passes through one of predefined zones. There are three zones located in the maze. They can not be sensed by an agent. The reward $\Delta_j \in \{0, 1, 2, 3\}$ is given by the number of zones visited in the j -th trial. The fitness value is then computed as $F = \sum_{j=1}^K \Delta_j + \sum_{j=1}^K \frac{S_j}{K}$, where $K = 2$ is the number of trials.

5 Experimental Results

Table 2 contains states with biggest and smallest Q-values and their best action. The states with biggest Q-values contain mostly perception NOWHERE. On the other side, states with smallest Q-values contain perception CRASHED.

Learned behaviour corresponds to obstacle avoidance behaviour. The most interested are the states, which contain perception "NEAR". Expected rules "when obstacle left, then turn right" can be found. States without perception "NEAR" were evaluated as safe — even if bad action was chosen in this state, it could be fixed by choosing good action in next state. Therefore, these actions do not tell us a lot about agent's behaviour. On the other side, action with perception VERYNEAR led to the crash, usually. Agent was not able to avoid the collision.

Table 2. 5 states with biggest and smallest Q-values and their best actions

left	State front	right	Action	Q-value
NOWHERE	NOWHERE	VERYFAR	[500, 300]	5775.71729
NOWHERE	NOWHERE	NOWHERE	[300, 300]	5768.35059
VERYFAR	NOWHERE	NOWHERE	[300, 500]	5759.31055
NOWHERE	NOWHERE	FEEL	[300, 300]	5753.71240
NOWHERE	VERYFAR	NOWHERE	[500, 100]	5718.16797
CRASHED	CRASHED	CRASHED	[300, 500]	-40055.38281
CRASHED	NOWHERE	CRASHED	[300, 300]	-40107.77734
NOWHERE	CRASHED	VERYNEAR	[300, 500]	-40128.28906
FAR	VERYNEAR	CRASHED	[300, 500]	-40210.53125
NOWHERE	CRASHED	NEAR	[200, 500]	-40376.87891

The experiment with evolutionary RBF network was repeated 10 times, each run lasted 200 generations. In all cases the successful behaviour was found, i.e. the evolved robot was able to explore the whole maze without crashing to the walls. Table 3 shows parameters of an evolved network with five RBF units. We can understand them as rules providing mapping from input sensor space to motor control. However, these 'rules' act in accord, since the whole network computes linear sum of the corresponding five Gaussians.

The following Tab. 4 shows rules from actual run of the robot in the train arena. The nine most frequently used rules are listed. It can be seen that this agent represents a typical evolved left-hand wall follower. Straight movement is a result of situations when there is a wall far left, or both far left and right. If the robot sees nothing, it rotates left-wise (rule 2). The front collision is avoided by turning right, as well as a near proximity to the left wall (rules 6–8).

The evolved robot was then tested in the bigger testing maze. It behaved in a consistent manner, using same rules, demonstrating generalization of the behaviour trained in the former maze.

Table 3. Rules represented by RBF units (listed values are original RBF network parameters after discretization)

left	Sensor		Width	Motor	
	front	right		left	right
VERYNEAR	NEAR	VERYFAR	1.56	500	-100
FEEL	NOWHERE	NOWHERE	1.93	-500	500
NEAR	NEAR	NOWHERE	0.75	500	-500
FEEL	NOWHERE	NEAR	0.29	500	-500
VERYFAR	NOWHERE	NOWHERE	0.16	500	500

Table 4. Most important rules represented by trained RBF network and their semantics

left	Sensor		Motor		
	front	right	left	right	
FEEL	NOWHERE	NOWHERE	500	500	straight forward
NOWHERE	NOWHERE	NOWHERE	100	500	turning left
VERYFAR	NOWHERE	NOWHERE	500	500	straight forward
FEEL	NOWHERE	FEEL	500	500	straight forward
NOWHERE	NOWHERE	FEEL	100	500	turning left
FAR	NOWHERE	NOWHERE	500	300	turning right
FEEL	FEEL	NOWHERE	500	300	turning right
NEAR	NOWHERE	NOWHERE	500	100	turning right

Both approaches were successful in finding the strategy for maze exploration. The 200 generations of evolutionary learning needed on average 8822 fitness evaluations (corresponds approx. 14 115 RL epochs). To acquire the desired behaviour, RBF networks needed from 529 to 2337 fitness evaluations.

6 Conclusion

We have presented experiments with RL and ER algorithms training a robot to explore a maze. It is known from the literature, and from our previous works, that this problem is manageable by both RL and ER learning with different neural representations of the control mechanism. Usually, in such a successful learning episode, an agent with general behavioural pattern emerges that is able to explore previously unseen maze in an efficient way.

In this work we have focused on comparison of rules derived by traditional RL approach and by the evolved neural networks. We have chosen the RBF network architecture with local processing units. These networks are easily to interpret in terms of rules than traditional perceptron networks. A simple analysis shows that both RL and ER resulted in a rules that are reasonable, and easy to interpret as higher-level behaviour traits. The RL approach shows rational obstacle avoidance, while the neuro-evolution approach comes with more compact individuals that can be clearly classified as left-hand wall followers (or right-hand wall followers, respectively).

As we have seen, different learning approaches can lead to different behaviours. Agents trained by evolutionary algorithms usually show simple behaviours. Often, changing basic environment constraints (dimensions of environment, for example) can make learned strategy fail [3]. In our experiment, learned strategy is simple (it can be described by several rules) but effective. Agent learned by Q-learning algorithm showed more complex behaviour. It can cope with situations, in which agent trained by ER would fail. However, effective wall following strategy was not discovered.

In our further work, we would like to take advantages of both approaches. The basic learning mechanism will be evolutionary algorithm. Behavioural diversity could be maintained by managing population of agents that use different learning approaches (RBF networks and reinforcement learning). In both algorithms used, experience can be expressed as a set of rules. Taking this into account, genetic operators could be designed to allow simple rules exchange mechanisms.

Acknowledgements. This research has been supported by the the project 1ET100300419 of the Program Information Society (of the Thematic Program II of the National Research Program of the Czech Republic) Intelligent Models, Algorithms, Methods and Tools for the Semantic Web Realization” and by the Institutional Research Plan AV0Z10300504 ”Computer Science for the Information Society: Models, Algorithms, Applications”.

References

1. Pfeifer, R., Scheier, C.: Understanding Intelligence. The MIT Press, Cambridge (2000)
2. Sutton, S.R., Barto, G.A.: Reinforcement Learning: An Introduction. The MIT Press, Cambridge (1998)
3. Nolfi, S., Floreano, D.: Evolutionary Robotics — The Biology, Intelligence and Technology of Self-Organizing Machines. The MIT Press, Cambridge (2000)
4. Holland, J.: Adaptation In Natural and Artificial Systems. MIT Press, Cambridge (1992)
5. Haykin, S.: Neural Networks: A Comprehensive Foundation, 2nd edn. Prentice Hall, Englewood Cliffs (1999)
6. Slušný, S., Neruda, R.: Evolving Homing Behaviour for Team of Robots. Computational Intelligence, Robotics and Autonomous Systems. Massey University, Palmerston North (2007)
7. Slušný, S., Neruda, R., Vidnerová, P.: Evolution of Simple Behavior Patterns for Autonomous Robotic Agent. System Science and Simulation in Engineering, pp. 411–417. WSEAS Press (2007)
8. Watkins, C.J.C.H.: Learning from Delayed Rewards. Ph.D. thesis (1989)
9. Mitchell, T.: Machine Learning. McGraw Hill, New York (1997)
10. Poggio, T., Girosi, F.: A Theory of Networks for Approximation and Learning. Technical report, A. I. Memo No. 1140, C.B.I.P. Paper No.31, Massachusetts Institute of Technology, Cambridge, MA, USA (1989)
11. Moody, J., Darken, C.: Fast Learning in Networks of Locally-tuned Processing Units. Neural Computation 1, 289–303 (1989)
12. Broomhead, D., Lowe, D.: Multivariable Functional Interpolation and Adaptive Networks. Complex Systems 2, 321–355 (1988)
13. Fogel, D.: Evolutionary Computation: The Fossil Record. MIT-IEEE Press (1998)
14. E-puck, online documentation, <http://www.e-puck.org>
15. Webots simulator on-line documentation, <http://www.cyberbotics.com/>

Self-Guided Genetic Algorithm

Shih-Hsin Chen¹, Pei-Chann Chang², and Qingfu Zhang³

¹ Department of Electronic Commerce Management, Nanhua University. 32, Chungkeng, Dalin, Chiayi 62248, Taiwan R.O.C.

² Department of Information Management, Yuan-Ze University. 135 Yuan-Dong Rd., Taoyuan 32026, Taiwan, R.O.C

³ Department of Computing and Electronics Systems, University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, U.K.

Abstract. This paper proposed a new algorithm, termed as Self-Guided genetic algorithm, which is one of the algorithms in the category of evolutionary algorithm based on probabilistic models (EAPM). Previous EAPM research explicitly used the probabilistic model from the parental distribution, then generated solutions by sampling from the probabilistic model without using genetic operators. Although EAPM is promising in solving different kinds of problems, Self-Guided GA doesn't intend to generate solution by the probabilistic model directly because the time-complexity is high when we solve combinatorial problems, particularly the sequencing ones. As a result, the probabilistic model serves as a fitness surrogate which estimates the fitness of the new solution beforehand in this research. So the probabilistic model is used to guide the evolutionary process of crossover and mutation. This research studied the single machine scheduling problems and the corresponding experiment were conducted. From the results, it shows that the Self-Guided GA outperformed other algorithms significantly. In addition, Self-Guided GA works more efficiently than previous EAPM. As a result, it could be a break-through in the branch of EAPM.

1 Introduction

Evolutionary Algorithm based on Probabilistic models (EAPM) is one of the most popular evolutionary algorithms in recent years [2][6][8]s. EAPM explicitly build a probabilistic model to model the parental distribution, and EAPM generates solutions by sampling from the probabilistic model whereas GA employs crossover and mutation operator. The reason why EAPM doesn't use genetic operators is that the problem-independent crossover operator may either break the building blocks of chromosomes or may not mix the genetic information properly [9]. This is the most important characteristic to distinguish EAPM and GA.

Previous EAPMs, such as EA/G [11], ant-miner [1], or ACGA [3], are used to generate the new solutions by sampling from the probabilistic model; however, the authors discovered that the proportional selection causes higher computational cost in solving sequencing problems. As a result, this research attempts to employ the probabilistic models to guide the evolutionary process in solving

the scheduling problems. The idea is that the probability of each item is further interpreted as a figure of merit of an item at a position in a solution, i.e., the probabilistic model serves as a fitness surrogate. Then, we can evaluate the goodness of the modified solutions by the probability depended on the probabilistic models. After we summarize the new probabilities and then differentiate the original probability, the differentiated value represents the prediction of the new solution and evolutionary algorithm decides whether they accept the new solution before we use the true fitness function. By using this idea of the probabilistic model as a fitness surrogate, we can embed this concept with different kinds of the operators which modify the existing solutions.

This research will utilize this concept to guide the crossover and mutation operator as an example. This concept is not limited to the genetic operators. Instead, this concept is able to be applied in different operators. This proposed algorithm, termed as Self-Guided Genetic Algorithm, is proposed in this research and it adopts the probabilistic model guiding the crossover and mutation operators. The proposed algorithm evaluates the probability differences of the genes varied by crossover and mutation operators. Furthermore, Self-Guided Genetic Algorithm selects the best alternative from some candidates when crossover or mutation varies the genes. So the characteristics of this algorithm are that the evolutionary direction is guided by the global information of probabilistic model without losing the location information and that it prevents GA from being a blind search.

The rest of the paper is organized as follows: Section 2 is the detail explanations of the Self-Guided GA, Section 3 is the experiment results whereas the proposed algorithm was evaluated by using the single machine scheduling problems, which considered the minimization of earliness/tardiness cost, Section 4 is the discussions and conclusions of this research.

2 Self-Guided Genetic Algorithm

EAPM extracts the gene variable structure from population distribution and expresses it by a probabilistic model [7]. This research will embed the probabilistic model into crossover and mutation operators which will guide the evolution progress towards more promising solution space. The probabilistic model serves as a fitness surrogate and the probabilistic model will evaluate the figure of merit beforehand. It means that the probabilistic model will calculate the probability difference of two selected genes located at different positions to guide the movement of genes decided by crossover and mutation operators. As a result, this proposed EAPM algorithm will guide the searching direction instead of blindly searching the solution space and it doesn't rely on the proportional selection to generate solutions.

The benefits of the proposed method are preserving the salient genes of the chromosomes, and exploring and exploiting good searching direction for genetic operators. In addition, since the probabilistic difference provides good neighborhood

information, it can serve as a fitness function surrogate. The detailed procedure of this self-guided genetic algorithm is described as follows:

Population: A set of solutions
generations: The maximum number of generations
P(t): Probabilistic model
t : Generation index

- 1: Initialize *Population*
- 2: $t \leftarrow 0$
- 3: Initialize *P(t)*
- 4: **while** $t < \text{generations}$ **do**
- 5: EvaluateFitness (*Population*)
- 6: Selection/Elitism(*Population*)
- 7: $P(t + 1) \leftarrow \text{BuildingProbabilityModel}(\text{Selected Chromosomes})$
- 8: Self-Guided Crossover()
- 9: Self-Guided Mutation()
- 10: $t \leftarrow t + 1$
- 11: **end while**

Fig. 1. Algorithm1: MainProcedure()

Step 2 initializes the probability matrix $P(t)$ by using $1/n$ where n is the problem size. Step 7 builds the probabilistic model $P(t)$ after the selection procedure. The details of $P(t)$ are explained in Section 2.1. In Step 8 and Step 9, $P(t)$ is employed in self-guided crossover operator and self-guided mutation operator. Probabilistic model will guide the evolution direction, which is shown in Section 2.2 and Section 2.3. In this research, two-point central crossover and swap mutation are applied in the crossover and mutation procedures in scheduling problems.

The following sections explain the proposed algorithm in details, which at first explain how to establish a probabilistic model and how the probabilistic model guides the crossover and mutation operators.

2.1 Establishing a Probabilistic Model

Suppose a population has M strings X^1, X^2, \dots, X^M at current generation t , which is denoted as Population(t). Then, X_{ij}^k is a binary variable in chromosome k . The fitness of these M chromosomes is evaluated and the gene information is collected from N best chromosomes where $N < M$. The N chromosomes are set as $M/2$ in this research. The purpose of selecting only N best chromosomes from population is to prevent the quality of the probabilistic model from being down-graded by inferior chromosomes. Let $P_{ij}(t)$ be the probability of job i to appear at position j at the current generation. As in PBIL, the probability $P_{ij}(t)$ is updated as follows:

$$P_{ij}(t + 1) = (1 - \lambda)P_{ij}(t) + \lambda \frac{1}{N} \sum_{k=1}^N X_{ij}^k, i = 1, \dots, n, j = 1, \dots, n \quad (1)$$

where $\lambda \in (0, 1]$ is the learning rate. The larger λ is, the more the gene information of the current population contributes. After we have $P_{ij}(t)$, the $P(t + 1)$ is also obtained. Therefore, the expectation fitness of a solution X^k can be estimated from the probabilistic matrix, by means of joint probability distribution shown as follows:

$$P_{t+1}(X^k) = P_{t+1}(X^k_{[1],1}, \dots, X^k_{[n],n}) \approx \prod_{p=1, i=[p]}^n P_{t+1}(X^k_{ip}) \tag{2}$$

Where $[p]$ is the job at Position p in Chromosome X^k .

In general, the procedures of establishing the probabilistic model include (1) Evaluating fitness of all M solutions, (2) Selecting N best solutions, and (3) Learning the joint distribution of the selected solutions. After a probabilistic model is built up, the probabilistic model can serve as a surrogate of fitness evaluation since the $\prod_{p=1, i=[p]}^n P_{t+1}(X^k_{ip})$ estimates the goodness of the solution X^k .

As soon as the probabilistic model $P(t + 1)$ simplified as P in the rest of sections is established, it is embedded into crossover and mutation operator. Since the concept of swap mutation is simple and well-known, self-guided mutation operator is introduced first to solve combinatorial problems based on the probabilistic model.

2.2 Mutation Operator with Probabilistic Model

Suppose two jobs i and j are randomly selected, which are located at position a and position b , respectively. Thus, p_{ia} and p_{jb} denote job i at position a , and job j at position b . After these two jobs are swapped, the new probabilities of the two jobs become p_{ib} and p_{ja} . The probability difference Δ_{ij} is calculated as Eq. 3, which is the partial evaluation of the probability difference because the probability sum of other jobs remains the same.

$$\Delta_{ij} = P(X') - P(X) = \prod_{p \notin \{a, b\}, g=[p]}^n P_{t+1}(X_{gp}) [(p_{ib}p_{ja}) - (p_{ia}p_{jb})] \tag{3}$$

Now that the $\prod_{p \notin \{a, b\}, g=[p]}^n P_{t+1}$ is always ≥ 0 , Eq. 3 is changed as:

$$\Delta_{ij} = (p_{ib}p_{ja}) - (p_{ia}p_{jb}) \tag{4}$$

Later on, the research observed the probability value appeared to be zero frequently in the combinatorial problems. When there is a probability value which is zero in Equation 4, it causes the multiplied probability, such as $p_{ib}p_{ja}$ or $p_{ia}p_{jb}$, to become zero. To overcome this problem, the equation 4 is slightly modified the $p_{ib}p_{ja}$ and $p_{ia}p_{jb}$ into $(p_{ib} + p_{ja})$ and $(p_{ia} + p_{jb})$. It is shown as follows:

$$\Delta_{ij} = (p_{ib} + p_{ja}) - (p_{ia} + p_{jb}) \tag{5}$$

If Δ_{ij} is positive, it implies that one gene or both genes might move to a promising area. On the other hand, while Δ_{ij} is negative, the implication means that at least one gene moves to an inferior position.

On the basis of probabilistic difference, it is natural to consider different choices of swapping points during the mutation procedure. A parameter TM is introduced for the self-guided mutation operator, which denotes the number of tournaments of comparing the probability differences among the TM choices in swap mutation. Basically, $TM \geq 2$ and $TM = 1$ implies that the mutation operator mutates the genes directly without comparing probability difference among the different TM choices.

When $TM = 2$, suppose the other alternative is that two jobs m and n are located at Position c and position d , respectively. The probability difference of exchanging job m and n is:

$$\Delta_{mn} = (p_{md} + p_{nc}) - (p_{mc} + p_{nd}) \quad (6)$$

After Δ_{ij} and Δ_{mn} are obtained, the difference of the two alternatives is as follows:

$$\Delta = \Delta_{ij} - \Delta_{mn} \quad (7)$$

If $\Delta < 0$, the contribution of swapping job m and n is better, so that we swap job m and n . Otherwise, job i and j are swapped. Consequently, the option of larger probability difference is selected and the corresponding two jobs are swapped. By observing the probability difference Δ , the self-guided mutation operator exploits solution space which enhances the solution quality and prevents from destroying some dominant genes on a chromosome. Moreover, the main procedure of self-guided mutation is Eq. 7 where the time-complexity is only constant time, which is the same with swap mutation. Thus, this approach is proved to be able to work efficiently.

About the setting of TM , a design-of-experiment (DOE) is conducted for different kinds of problems which are shown in experimental results. The result shows that TM is generally set as 2.

Extended from the probabilistic model for mutation operator, the next section continues the idea of utilizing probability difference which is embedded into crossover operator.

2.3 Crossover Operator with Probabilistic Model

The concept of mutation is to exploit local information while the crossover does global search by mating two chromosomes. Thus, the idea of self-guided crossover approach attempts to mate a chromosome with the other appropriate chromosome, which yielded better offspring when the cutting points of two-point crossover were given in advance. Similar to TM in the self-guided mutation operator, TC is the number of tournament selection of different crossover options. The evaluation of the TC choices is also done in the same way that we calculate the probability difference of the genes moving to different positions. Due to the

probabilistic model which serves as a fitness surrogate, we are able to estimate which chromosome is a better mating choice that may improve the fitness of the new offspring better.

Because this research employs the two-point center crossover, the following description is based on this crossover method, which the sequence between the two cut points are altered. Suppose the two cut-points are named CP1 and CP2 and there is a parent solution named Parent1 which will be mated with a solution. Two solutions, named Candidate1 and Candidate2, is randomly selected to be mated with a Parent1. By the Eq. 8, we can determine which new offspring should be adopted which is depended on the expected fitness of the two offsprings. If the value is positive, it means we should select Candidate1; instead, we should select Candidate2 to be mated with Parent1.

$$\Delta = \prod_{p \in (CP1 \text{ to } CP2), g=[p]}^n P(Candidate1_{gp}) - \prod_{p \in (CP1 \text{ to } CP2), g=[p]}^n P(Candidate2_{gp}) \quad (8)$$

To conclude the Self-Guided GA, this algorithm is obviously different from the previous EAPM algorithms which explicitly samples new solutions without using the crossover and mutation operators. Self-Guided GA, instead, embeds the probabilistic model into crossover operator and mutation operator, which explore and exploit solution space, respectively. Most important of all, the time-complexity of the algorithm works more efficiently than previous EAPM ones in solving sequencing problems. This research, thus, placed the Self-Guided GA as a new branch in the field of EAPM.

3 Experiment Results

In order to evaluate the performance of the Self-Guided GA, it was compared with some algorithms in literature, such as Genetic Algorithm with Dominance Properties (GADP), Artificial Chromosome with Genetic Algorithm (ACGA), the combination of ACGA and GADP (ACGADP), ACGA with evaporation [5], and Guided Mutation (EA/G) which can be found in [4], [3], and [11]. These algorithms are used to test the single machine scheduling problems and these instances are taken from [10]. This single machine scheduling problems with the consideration of minimizing the earliness and tardiness were evaluated. These algorithms were implemented by Java 2 (With JBuilder JIT compiler) on Windows 2003 server (Intel Xeon 3.2 GHZ). In all the experiments, each instance was replicated 30 times. Because of the page limits to show the complete statistics results in this paper, please refer to our website [6].

To verify the difference of the algorithms, ANOVA is employed. Because there is significant difference of among methods, in order to justify the performance of these algorithms, Duncan pair-wise comparison is employed in Table 1, which shows that there is a significant difference between/among subjects if they share

¹ <http://ppc.iem.yzu.edu.tw/File/sourceCodes/InjectionArtificialChromosomes/>

different alphabet. Otherwise, there are no differences between/among the subjects. In Table 1, Duncan comparisons indicated Self-Guided GA which performed as well as ACGA with evaporation, ACGA, EA/G, and ACGADP . On the other hand, GADP is the second group in this comparisons and SGA is the worst in the single machine scheduling problems.

Table 1. Duncan Grouping in testing objective values of single machine scheduling problems

Duncan Grouping	Mean	N	method
A	13982.894	6420	SGA
B	12827.096	6420	GADP
C	12816.471	6420	ACGADP
C			
C	12813.66	6420	EA/G
C			
C	12813.276	6420	ACGA
C			
C	12813.253	6420	Self-Guided GA
C			
C	12811.868	6420	ACGAEvaporation

After we tested the ANOVA results in minimizing objective values, the computational time is also examined and the corresponding results are shown on our website. Since the CPU time of methods are significant, Duncan Grouping results indicated that there is no significance between SGA and Self-Guided GA and both of them work more efficient than others, particularly the ACGA and EA/G. Consequently, these results show that Self-Guided GA is very attractive because the algorithm performs well and it is not time-consuming compared with Standard Genetic Algorithm.

4 Discussions and Conclusion

The paper unveils a new concept of EAPMs, which is to apply the probabilistic model evaluating the figure of merit of a new solution beforehand. So the probabilistic model is able to guide the evolution process instead of using sampling from probabilistic model to generate solutions. Although this concept is able to be employed in many aspects, this paper embedded this concept with crossover and mutation operator as an example. As a result, Self-Guided GA enables the GA out of blindly searching and the proposed algorithm works efficiently than previous EAPMs. The proposed algorithm solved single machine scheduling problems with minimization of earliness/tardiness cost. The experiment result indicated Self-Guided GA indeed performed well when it is compared with some algorithms in literature. In addition, the proposed algorithm works as efficient as SGA. It is because the evaluation of the probabilistic difference

of crossover and mutation operators takes only $O(n)$ and a constant time, respectively. The extra time-complexity of this operation is dominated by that of crossover operator. When it comes to previous EAPMs, the time complexity is $O(n^2)$. Thus, this approach doesn't lead to extra or excessive computational efforts. Based on this pioneer research, researchers are able to design an operator which integrates the proposed concept in the near future.

References

1. Aickelin, U., Burke, E.K., Li, J.: An Estimation of Distribution Algorithm with Intelligent Local Search for Rule-based Nurse Rostering. *Journal of the Operational Research Society* 58(12), 1574–1585 (2007)
2. Baraglia, R., Hidalgo, J., Perego, R., Cnuce, I., Cnr, P.: A Hybrid Heuristic for the Traveling Salesman Problem. *IEEE Transactions on Evolutionary Computation* 5(6), 613–622 (2001)
3. Chang, P., Chen, S., Fan, C.: Mining Gene Structures to Inject Artificial Chromosomes for Genetic Algorithm in Single Machine Scheduling Problems. *Applied Soft Computing Journal* 8(1), 767–777 (2008)
4. Chang, P., Chen, S., Mani, V.: A Hybrid Genetic Algorithm with Dominance Properties for Single Machine Scheduling with Distinct Due Dates. *Applied Mathematical Modeling* (2008), doi:10.1016/j.apm.2008.01.006
5. Chang, P., Chen, S., Fan, C.: Generating Artificial Chromosomes by Mining Gene Structures with Diversity Preservation for Scheduling Problems. *Annals of Operations Research* (accepted, 2008)
6. Harik, G., Lobo, F., Goldberg, D.: The Compact Genetic Algorithm. *IEEE Transactions on Evolutionary Computation* 3(4), 287–297 (1999)
7. Larranaga, P., Lozano, J.: Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation. In: Ebeling, W., Rechenberg, I., Voigt, H.-M., Schwefel, H.-P. (eds.) *PPSN 1996. LNCS*, vol. 1141, pp. 178–187. Springer, Heidelberg (1996)
8. Muhlenbein, H., Paaß, G.: From Recombination of Genes to the Estimation of Distributions I. Binary parameters. Kluwer Academic Publishers, Dordrecht (2002)
9. Pelikan, M., Goldberg, D., Lobo, F.: A Survey of Optimization by Building and Using Probabilistic Models. *Computational Optimization and Applications* 21(1), 5–20 (2002)
10. Sourd, F., Kedad-Sidhoum, S.: The One-Machine Problem with Earliness and Tardiness Penalties. *Journal of Scheduling* 6(6), 533–549 (2003)
11. Zhang, Q., Sun, J., Tsang, E.: An Evolutionary Algorithm With Guided Mutation for the Maximum Clique Problem. *IEEE Transactions on Evolutionary Computation* 9(2), 192–200 (2005)

Study on Discretization in Rough Set Via Modified Quantum Genetic Algorithm

Shuhong Chen and Xiaofeng Yuan

NO.1017 Calibration Lab of Beijing Region, Beijing, 100085, P.R. China
Chenyshuhong@yahoo.com.cn

Abstract. Optimal discretization of real valued attributes in rough set is a problem of NP-complete. To resolve this problem, a modified quantum genetic algorithm (MQGA) and a new parametric configuration scheme for the fitness function are proposed in this paper. In MQGA, a novel technique with locally hierarchical search ability is introduced to speed up the convergence of QGA. With this configuration scheme, it is convenient to distinguish the appropriate solutions that partition the new decision table consistently from all the results. Experiments on dataset of Iris have demonstrated that the proposed MQGA is more preferable compared with the traditional GA-based method and QGA based method in terms of execution time and ability to obtain the optimal solution.

Keywords: Quantum genetic algorithm; rough set; decision system; discretization.

1 Introduction

Rough set theory, introduced by Z. Pawlak in the early 1980s [1], is a new mathematical tool to deal with vagueness and uncertainty. This approach has found applications in the areas of machine learning, knowledge acquisition, decision analysis, knowledge discovery from databases, expert systems, decision support systems, inductive reasoning, and pattern recognition [2].

Optimal discretization of real valued attributes in rough set, is a problem of NP-complete, and there has been much research effort devoted to it in recent years. According to the impact on consistency of decision table, the problem can be grouped into two classes, A and B. In class A, which includes the Boolean Reasoning Approach (BRA) [3], the BRA based greedy algorithm [4], and the GA based approach [5], discretization is performed without changing the consistency of the decision table. On the contrary, consistency is not guaranteed after discretization in class B, which includes the Chi2 algorithm [6], the hierarchical clustering method [7], and information entropy based method [8]. With the development in evolutionary computing and quantum computing, a quantum genetic algorithm [9], which is characterized by principles of quantum computing, was proposed in 2000. Just like genetic algorithms, QGA is a probabilistic algorithm. It is validated experimentally that the QGA outperforms the conventional GA on the well-known combinatorial optimization problem, namely, the knapsack problem, due to its special designed qubit chromosomes and updating mechanisms.

Motivated by the emergence of QGA, the authors proposed a modified quantum genetic algorithm (MQGA) based method for performing discretization in rough set. In MQGA, a novel locally hierarchical search technique is introduced to speed up the convergence process. Comparative experiments on Iris dataset have demonstrated that proposed method has the capability of achieving the consistent and minimal discretization of decision system, and outperforms the conventional GA based method in terms of efficiency and solution quality.

The rest of the paper is organized as follows. Section 2 gives an introduction of the discretization and its theoretical basis. Section 3 gives the overall description of QGA. Section 4 proposes the modified QGA and its detailed application procedure in discretization. Section 5 illustrates some experimental results of QGA, modified QGA, and conventional GA as well. Finally, section 6 presents the concluding remarks.

2 Preliminaries for Discretization

2.1 Description of Discretization

A decision table $S = \langle U, A \cup \{d\}, V, f \rangle$, is described as follows:

U: a finite set with N objects, and $U = \{u_1, u_2, \dots, u_N\}$; **A**: a set of k condition attributes, $A = \{a_1, a_2, \dots, a_k\}$; **d**: denotes a decision attribute; $A \cap \{d\} = \emptyset$; **V**: denotes the value domain of condition attribute, and $V = \bigcup_{a \in A} V_a$, $V_a = [l_a, r_a] \subset \mathbb{R}$, where \mathbb{R} is the set of real numbers; **f**: $U \times (A \cup \{d\}) \rightarrow V$, is a information function.

Definition 1: Any pair (a, c) , where $a \in A$, and $c \in \mathbb{R}$, defines a partition of V_a by a left-hand-side and right-hand-side interval. And the pair (a, c) is called a cut on V_a [5].

Therefore, for an attribute $a \in A$, a partition on V_a can be defined by a set of cuts, as

$$D_a = \{ (a, c_1^a), (a, c_2^a), \dots, (a, c_{k_a}^a) \}. \tag{1}$$

where $k_a \in \mathbb{N}$, and $l_a = c_0^a < c_1^a < c_2^a < \dots < c_{k_a}^a < c_{k_a+1}^a = r_a$. According to the cut set, V_a can be redefined as a union of sub-intervals. For instance,

$$V_a = [c_0^a, c_1^a] \cup [c_1^a, c_2^a] \cup \dots \cup [c_{k_a}^a, c_{k_a+1}^a]. \tag{2}$$

Accordingly, any set of cuts $D = \bigcup_{a \in A} D^a$ can transform the original continuous decision table into a discrete one, $S^D = \langle U, A \cup \{d\}, V^D, f^D \rangle$, where $f^D(u, a) = i$ is the counterpart of $f(u, a) \in [c_i^a, c_{i+1}^a]$, $i \in \{0, 1, \dots, k_a\}$, and $u \in U, a \in A$.

The task of discretization is to determine a minimal set of cuts for the transformed decision table S^D without changing discernability.

The widely accepted criteria for discretization rationality evaluation, is as follows [10]:

- (a) Consistency of D : any two objects discerned by A should be discerned by D as well.
- (b) Minimum: there is no $D' \subset D$, satisfying consistency.
- (c) Optimality: if D' follows the inequality $\text{card}(D') \cong \text{card}(D)$ with consistency, then it is an optimal set of cuts.

2.2 Determination of Candidate Cut Set

For a decision system S, if $a(u_i)=v_i^a$, $a \in A$, $v_i^a \in V_a$, $u_i \in U$; $v_1^a < v_2^a < \dots < v_{n_a}^a$; then the maximum cut set on a is as

$$C_a = \{ (a, \frac{v_1^a + v_2^a}{2}), (a, \frac{v_2^a + v_3^a}{2}), \dots, (a, \frac{v_{n_a-1}^a + v_{n_a}^a}{2}) \} \cdot \tag{3}$$

Therefore, the maximum cut set on A is as

$$C_A = \cup_{a \in A} C_a \tag{4}$$

Definition 2: The cut (a, c) is called a bound cut only if the followings are satisfied:

- (a) There exist $u_i, u_j \in U$, $d(u_i) \neq d(u_j)$, satisfying $a(u_i) < c < a(u_j)$;
- (b) There is no $u \in U$, satisfying $a(u_i) < a(u) < a(u_j)$.

Based on Definition 2 and equation (4), an initial candidate cut set is obtained, which is comprised of the bound cuts. With the initial candidate cut set, a new decision table S^D can be constructed. For detailed information, refers to [11].

3 Quantum Genetic Algorithm

3.1 Representation

Based on the concept of quantum and superposition of states of quantum mechanics, the smallest unit of information in QGA is a quantum bit or qubit, which may be in the ‘1’ state, in the ‘0’ state, or in any superposition of the two. The state of qubit can be expressed as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where α and β are complex numbers, and denote the probability amplitudes of the current states. $|\alpha|^2$ and $|\beta|^2$ represent the probability that the quit assumes the ‘0’ state and ‘1’ state, respectively, and the equation $|\alpha|^2 + |\beta|^2 = 1$ is satisfied. In QGA, a chromosome is comprised of n qubits, and it can be expressed as

$$q_j^i = \begin{pmatrix} \alpha_1^i & \alpha_2^i & \dots & \alpha_n^i \\ \beta_1^i & \beta_2^i & \dots & \beta_n^i \end{pmatrix} \tag{5}$$

where $|\alpha_i|^2 + |\beta_i|^2 = 1$, $i=1, 2, \dots, n$. The symbol t and j are the sequence numbers of the current generation and chromosome respectively. Under the framework of quantum mechanics, this chromosome can represent 2^n states at one time, until it collapses to a single state by observation.

3.2 Procedure of QGA

Generally, the procedure of QGA can be itemized as following steps:

Step1 initialize Q(t). Initialize the population $Q(t) = \{q_1^t, q_2^t, \dots, q_m^t\}$, and the starting generation $t=0$. The probability amplitudes of each qubit, α_{ij} and β_{ij} , are initialized to $1/\sqrt{2}$ and $1/\sqrt{2}$.

Step2 observe Q(t). Based on observation of the population $Q(t)$, the result $C(t) = \{c_1^t, c_2^t, \dots, c_m^t\}$ is obtained, where c_j^t denotes a binary string of length n. The

string is generated by comparing the probability amplitude $|\alpha_i|^2$ (or $|\beta_i|^2$) of each qubit with a random number r , which ranges from 0 to 1. Picking up a random number r , if $|\alpha_i|^2 < r$ (or $|\beta_i|^2 < r$) is satisfied; the corresponding qubit is set to '1', otherwise, '0'.

Step3 evaluate C(t). Each fitness of observation value in C(t) is calculated by invoking the fitness function.

Step4 store the best solution. According to fitness of each observation value, pick the best solution in Q(t), and store it in B(t). If B(t) has kept unchanged for I_g consecutive generations, the Q(t) should be updated by a special cursive operation [12], which generates a new population with random probability amplitudes.

Step5 termination judgement. If the termination term is satisfied, stop the procedure.

Step6 update Q(t) with quantum gate. The quantum gates, which are usually designed corresponding to different applications, generate the source power of evolution. And, a quantum gate can be expressed as

$$G(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}. \tag{6}$$

where θ is a rotation angle. For a chromosome, $[\alpha_i \ \beta_i]^T$, the probability amplitude pair of the i -th qubit, can be adjusted to $[\alpha'_i \ \beta'_i]^T$ through the quantum gate, as in (7).

$$\begin{bmatrix} \alpha'_i \\ \beta'_i \end{bmatrix} = G(\theta_i) \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \\ \sin \theta_i & \cos \theta_i \end{bmatrix} \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix}. \tag{7}$$

$$\theta_i = s(\alpha_i, \beta_i) \Delta\theta_i. \tag{8}$$

where θ_i is a rotation angle. The function $s(\alpha_i, \beta_i)$ has three possible values, 1, 0, or -1. The evolutionary speed is determined by rotation step $\Delta\theta_i$, which usually ranges from 0.05π to 0.0025π . A typical lookup table of rotation angle is as Table 1.

Table 1. Lookup table of $s(\alpha_i, \beta_i)$

x_i	b_i	$f(x) \leq f(b)$	$s(\alpha_i, \beta_i)$			
			$\alpha_i \beta_i > 0$	$\alpha_i \beta_i < 0$	$\alpha_i = 0$	$\beta_i = 0$
0	0	False	0	0	0	0
0	0	True	0	0	0	0
0	1	False	+1	-1	0	± 1
0	1	True	-1	+1	± 1	0
1	0	False	-1	+1	± 1	0
1	0	True	+1	-1	0	± 1
1	1	False	0	0	0	0
1	1	True	0	0	0	0

where x_i and b_i are the i -th binary bit of the binary solution x and b , respectively. Symbol b denotes the best solution by far. The symbol $f(\cdot)$ denotes the fitness function.

Step7 return to Step2 and continue.

4 Modified QGA for Discretization

4.1 Cut Set Representation and Fitness Function

A cut set is represented by a chromosome, for instance, if maximum cut set is 8, then the length of chromosome is set to 8. A chromosome ‘01010100’ represents a cut set with 3 cuts, which are marked by the bit of ‘1’.

A well-designed fitness function is of vital importance. In this connection, a fitness function is constructed elaborately as below

$$f(x) = (1 - \lambda) (1 - \text{card}(x) / n) + \lambda \text{card}(D) / \text{card}(T) . \tag{9}$$

Where x is a solution represented by a quantum chromosome; λ is a regulating factor; and n is the length of chromosome. $\text{Card}(x)$ denotes the number of state ‘1’ in the binary representation of x . $\text{Card}(D)$ represents the number of object pairs that x can discern in the new decision table S^D . $\text{Card}(T)$ denotes the number of object pairs in the new decision table S^D . For $f(x)$, the first part is designed to find a minimum cut set; the second part is to guarantee the consistency of discretization.

Lemma: For the fitness function (9), if the original decisional table is consistent, and λ can be described as (10); then the sufficient and necessary condition, which guarantees a consistent discretization by x , is that the inequation $f(x) > \lambda$ is existent.

$$1 > \lambda > \text{card}(T) (n-1) / [n + \text{card}(T) (n - 1)] . \tag{10}$$

Proof: Assume that the discretization result of x is inconsistent under the condition of $f(x) > \lambda$. Based on the definitions of the fitness function and consistency, the equation below exists.

$$\max\{ f(x) \} = (1 - \lambda) (1 - 1 / n) + \lambda [\text{card}(T) - 1] / \text{card}(T) \tag{11}$$

Combined (11) with (10), we have the inequation $\max\{f(x)\} < \lambda$, which is in contradiction with the assumption inequation $f(x) > \lambda$. Therefore, when $f(x) > \lambda$ is existent, the discretization by x is consistent.

Conversely, according to the definition of consistency, if the discretization result is consistent, then the equation below is satisfied

$$\text{card}(D) / \text{card}(T) = 1 . \tag{12}$$

Then combined (12) with the first part of $f(x)$, as

$$(1 - \lambda) (1 - \text{card}(x) / n) > 0 . \tag{13}$$

It can be concluded that $f(x) > \lambda$. Therefore, if the discretization result is consistent, then $f(x) > \lambda$ is existent. The proof is completed.

Based on the lemma, λ can be chosen properly; accordingly, the consistency of discretization result can be deduced conveniently from $f(x)$.

4.2 Locally Hierarchical Search Technique and Modified QGA

Although, QGA has advantages over the conventional GA in terms of population diversity and global search ability, convergence performance of QGA requires further

study, especially when the length of chromosome is large. Motivated by improving convergence performance of QGA, a locally hierarchical search technique (LHST) is proposed. In LHST, the discernability of each cut, which means how many object pairs can be discerned by one cut, provides heuristic information for the locally hierarchical search on corresponding set of object pairs in the new decision table S^D . With the heuristic information, LHST search the new decision table hierarchically, and finally find out the locally optimal cuts.

If a predefined condition is satisfied, then the procedure of LHST is started as follows

Step1: the set of current object pairs is initially all object pairs in S^D ;

Step2: from the current optimal cut set, select a cut with maximum discernability on the set of current object pairs;

Step3: if the cut can discern all pairs in the current set; then stop, else continue;

Step4: refresh the set of current object pairs by all the object pairs that cannot be discerned on set of the object pairs;

Step5: go to step2

Accordingly, the procedure of the modified QGA is as below,

Begin initialize $Q(t)$

While (not termination condition) do

{ Step1 observe $Q(t)$

Step2 evaluate $C(t)$

Step3 store the best solution

Step4 If a predefined condition is satisfied, then run LHST

Step5 update $Q(t)$ with quantum gates, return to step2 and continue.

}

End

5 Performance Tests

To test the performance the propose algorithm, experiments were performed on a PC platform with a 1.7GHz CPU. Professor Fisher’s Iris dataset [13], which includes 150 objects with 4 different attributes, is adopted in the tests.

According to the approach mentioned in chapter 2.2, there are 19, 13, 5 and 5 candidate cuts for the continuous attribute sepal-length, sepal-width, petal-length and petal-width, respectively. As a result, a new decision table with 7500 object pairs and 42 candidate cuts is obtained. There are 50 times of random tests are conducted, and experimental parameters and results are illustrated in Table 2 and Table 3 respectively. In Table 2, P_c and P_m denote the crossover rate and mutation rate respectively.

Table 2. Experimental parameters for GA, QGA and MQGA

	Initial population	Chromosome length	λ	$\Delta\theta_i$	I_g	P_c	P_m	Maximum generations
GA	20	42	0.99991	Null	Null	0.7	0.01	500
QGA	5	42	0.99991	0.05	100	Null	Null	500
MQGA	5	42	0.99991	0.05	100	Null	Null	500

Table 3. Results of GA, QGA and MQGA approach

	Card(x_b)=6	Card(x_b)=7	Card(x_b)=8	Card(x_b)=9	Mean Execution Time
GA	11	30	5	4	30.85 sec
QGA	24	18	8	0	11.43 sec
MQGA	28	20	2	0	11.51 sec

It has been verified based on the lemma in chapter 4.2, that all the test results can achieve a consistent partition on the new decision table. And the optimal set of cuts is {Sepal-length=6.85, Sepal-width=3.15, Petal-length=(2.45, 4.95), Petal-width=(1.55 1.75)}. From Table 3, it can be concluded that MQGA is best among the three approaches in terms of the capability of converging to the optimal solution and execution time.

6 Conclusion

To improve the performance of discretization in rough set, a modified QGA and a new parametric configuration scheme for the fitness function are proposed in this paper. It has been verified experimentally that the proposed MQGA has advantages over GA and QGA in terms of efficiency and solution quality. The MQGA exhibits some excellent characteristics in convergence in the tests. It is very meaningful to introduce the newly emerged QGA into the area of data mining, though there are some application details to be further studied in the future.

References

1. Dubois, D., Prade, H.F.: *Rough sets: Theoretical Aspects of Reasoning about Data*, by Z. Pawlak. Kluwer, Dordrecht (1991)
2. Zdzislaw, P., Jerzy, G., Roman, S.: *Rough Sets*. Communications of the ACM 38(6) (June 1995)
3. Nguyen, H.S., Skowron, A.: Quantization of Real Value Attributes, Rough Set and Boolean Reasoning Approaches. In: *Proceedings of the 2nd Joint Annual Conference on Information Science*, Wrightsville Beach, NC, pp. 34–37 (1995)
4. Hou, L.J., Wang, G.Y., Nie, N.: Discretization in Rough Set Theory. *Computer Science*, 89–94 (2000)
5. Dai, J.H., Li, Y.X.: Study on Discretization Based on Rough Set Theory. In: *Proceedings of the 1st International Conference on Machine Learning and Cybernetics*, Beijing, pp. 1371–1373 (2002)
6. Tay, E.H., Shen, L.: A Modified Chi2 Algorithm for Discretization. *IEEE Transactions on Knowledge and Data Engineering*, pp. 666–670 (2002)
7. Li, M., Wu, C.D., Han, Z.H.: A Hierarchical Clustering Method for Attribute Discretization in Rough Set Theory. In: *Proceedings of the 3rd International Conference on Machine Learning and Cybernetics*, Shanghai, pp. 3650–3654 (2004)
8. Xie, H., Cheng, H.Z., Niu, D.X.: Discretization of Continuous Attributes in Rough Set Theory Based on Information Entropy. *Chinese Journal of Computers* 28(9), 1571–1574 (2005)

9. Han, K.H., Kim, J.H.: Genetic Quantum Algorithm and Its Application to Combinatorial Optimization Problem [A]. In: Proceedings of the 2000 IEEE Congress on Evolutionary Computation [C], pp. 1354–1360 (2000)
10. Nguyen, H.S.: Discretization of Real Value Attributes: Boolean Reasoning Approach [Ph.D Dissertation]. Warsaw University, Poland (1997)
11. Dai, J.H., Li, Y.X.: The Application of Genetic Algorithm for Discretization of Decision System. *Microelectronics & Computer* 20(2), 19–21 (2003)
12. Ge-Xiang, Z., Wei-Dong, J., Lai-Zhao, H.: Feature Selection Algorithm Based Quantum Genetic Algorithm. *Control Theory & Applications* 22(5), 810–813 (2005)
13. <http://www.fmt.vein.hu/softcomp/ucidata/dataset.htm> (2006)

A Class of Random Fuzzy Programming and Its Hybrid PSO Algorithm

Yankui Liu, Xuejie Bai, and Fang-Fang Hao

College of Mathematics & Computer Science, Hebei University
Baoding 071002, Hebei, China
yliu@hbu.cn, baixuejie123@sina.com, hff29@163.com

Abstract. This paper presents a new class of two-stage random fuzzy programming with recourse (RFPR) problems. Since the RFPR problem usually includes random fuzzy parameters with infinite supports, it is inherently an infinite dimensional optimization problem that can rarely be solved directly by the conventional optimization algorithms. To overcome this difficulty, this paper developed an approximation method for the original RFPR problem, and turn it into a finite-dimensional one. We also establish a convergence relation between the objective values of the original problem and its approximating problem. To solve a general RFPR problem, we design a hybrid algorithm by integrating the approximation method, neural network (NN) and particle swarm optimization (PSO) algorithm. Finally, one numerical example is presented to demonstrate the effectiveness of the designed algorithm.

1 Introduction

In a complex decision-making process, we often face a hybrid uncertain environment in which both randomness and fuzziness coexist. To deal with this twofold uncertainty, it is required to combine credibility theory [2,6,7,8,9,11,13,16] and classical probability theory. Random fuzzy variable is an appropriate tool to describe this twofold uncertain phenomenon, it was defined as a map from a credibility space to a collection of random variables [7]. Based on random fuzzy theory [8], random fuzzy programming has been well developed in the literature. For example, Guo *et al.* [3] proposed a repair model in terms of random variable distributions with a fuzzy parameter. Liu and Liu [14] defined the mean chance and equilibrium chance of a random fuzzy event, and established three types of chance-constrained programming models. Liu and Liu [15] introduced the expected value operator of a random fuzzy variable, and developed three classes of random fuzzy expected value models.

Motivated by two-stage stochastic program [1] and fuzzy programming with recourse problems [10,17,18], the purpose of this paper is to study a new class of two-stage random fuzzy programming with recourse (RFPR) problems. The two-stage RFPR arises as an optimization problem in situations where some parameters of the underlying model are not known with certainty and characterized by random fuzzy variables with known possibility and probability distributions,

and the objective is to formulate an optimization problem via expectation criteria. It is assumed that some decisions must be taken before the outcome of random fuzzy parameters is revealed and thus must be based on the knowledge of the distribution of the parameters only. This is referred to as the first stage. In the second stage, outcomes of all random fuzzy parameters have been observed and some recourse (or corrective) actions may be taken. This setting corresponds to a variety of practical optimization problems under uncertainty. Due to the original two-stage RFPR is inherently an infinite-dimensional optimization problem that usually cannot be solved directly via the conventional optimization algorithms. To overcome this difficulty, this paper will discuss the approximation method for the original two-stage RFPR problem, in which we adopt the convergence modes for random fuzzy variables defined in [12].

The paper is organized as follows. Section 2 presents the formulation of a new class of two-stage RFPR problems. In Section 3, we develop an approximation method to the original two-stage RFPR problem, and turn it into a finite-dimensional one. We also discuss the convergence of the objective value of the original two-stage RFPR problem to that of the approximating two-stage problem. In Section 4, we design a hybrid algorithm by combining the approximation method, NN and a PSO algorithm [4,5] to solve the approximating RFPR problem. The effectiveness of the designed algorithm is demonstrated via one numerical example. Finally, Section 5 gives the conclusions.

2 Formulation of Two-Stage RFPR Problem

A two-stage random fuzzy programming is formally build as

$$\min_x \{c^T x + Q_E(x) \mid x \in D\} \tag{1}$$

where $D = \{x \in \mathbb{R}^{n_1} \mid Ax = b, x \geq 0\}$, $Q_E(x) = E_\xi[Q(x, \xi)]$, and $Q(x, \xi_\gamma(\omega))$ is the optimal value of the following linear programming

$$\begin{cases} \min & q^T(\xi_\gamma(\omega))y \\ \text{subject to} & W(\xi_\gamma(\omega))y = h(\xi_\gamma(\omega)) - T(\xi_\gamma(\omega))x \\ & y \geq 0. \end{cases} \tag{2}$$

Here E refers to the expected value operator of random fuzzy variable [15]. Some components of q, h, T and W in the above problem are random fuzzy parameters represented through the following affine sums

$$\begin{cases} q(\xi) = q^0 + \sum_{i=1}^r q^i \cdot \xi_i \\ h(\xi) = h^0 + \sum_{i=1}^r h^i \cdot \xi_i \\ T(\xi) = T^0 + \sum_{i=1}^r T^i \cdot \xi_i \\ W(\xi) = W^0 + \sum_{i=1}^r W^i \cdot \xi_i \end{cases}$$

where $\xi = (\xi_1, \xi_2, \dots, \xi_r)^T$ is a random fuzzy vector, and $q^i, h^i, T^i, W^i, i = 0, 1, \dots, r$ are deterministic matrices of $n_2 \times 1, m_2 \times 1, m_2 \times n_1, m_2 \times n_2$, respectively.

3 Approximating Two-Stage RFPR Problem

Suppose that $\xi = (\xi_1, \xi_2, \dots, \xi_r)^T$ is the random fuzzy parameter involved in the RFPR problem (II) with a support $\Xi \subset \mathfrak{R}^r$. This section is devoted to the approximation approach to ξ by finitely supported random fuzzy vectors so that the recourse function

$$Q_E(x) = E_\xi[Q(x, \xi)] \tag{3}$$

can be computed at each feasible decision x .

Suppose that $\xi = (\xi_1, \dots, \xi_r)^T$ is a continuous random fuzzy vector, i.e., for each $\gamma \in \Gamma$, ξ_γ is a continuous stochastic vector, and the support of ξ is denoted by $\Xi = \prod_{i=1}^r [a_i, b_i]$, where $[a_i, b_i]$ is the support of ξ_i for $i = 1, 2, \dots, r$. We adopt the following method to approximate ξ by a sequence $\{\zeta_m\}$ of finitely supported primitive random fuzzy vectors. The method can be described as follows.

For each integer m , $\zeta_m = (\zeta_{m,1}, \zeta_{m,2}, \dots, \zeta_{m,r})^T$ is constructed by the following method.

For each $i \in \{1, 2, \dots, r\}$, define $\zeta_{m,i} = g_{m,i}(\xi_i)$ for $m = 1, 2, \dots$, where the functions $g_{m,i}$'s are as follows

$$g_{m,i}(u_i) = \begin{cases} a_i, & u_i \in [a_i, a_i + \frac{1}{m}) \\ \sup \{ \frac{k_i}{m} \mid k_i \in Z, \text{ s.t. } \frac{k_i}{m} \leq u_i \}, & u_i \in [a_i + \frac{1}{m}, b_i] \end{cases}$$

and Z is the set of integers. Since $g_{m,i}$'s are Borel measurable functions, ζ_m is a random fuzzy vector for each m .

By the definition of $\zeta_{m,i}$, random fuzzy variable ξ_i takes on values in its infinite support $[a_i, b_i]$, while $\zeta_{m,i}$ takes on values a_i and k_i/m for $k_i = [ma_i] + 1, \dots, K_i$, where $[r]$ is the integer part of the real number r , $K_i = mb_i - 1$ or $[mb_i]$ according as mb_i is an integer or not an integer. In addition, for each k_i , as ξ_i takes its values in $[k_i/m, (k_i + 1)/m)$, random fuzzy variable $\zeta_{m,i}$ takes on the value k_i/m only. Consequently, for each $(\gamma, \omega) \in \Gamma \times \Omega$, we have

$$\xi_{i,\gamma}(\omega) - \frac{1}{m} < \zeta_{m,i,\gamma}(\omega) \leq \xi_{i,\gamma}(\omega), \quad i = 1, 2, \dots, r$$

which leads to

$$\|\zeta_{m,\gamma}(\omega) - \xi_\gamma(\omega)\| = \sqrt{\sum_{i=1}^r (\zeta_{m,i,\gamma}(\omega) - \xi_{i,\gamma}(\omega))^2} \leq \frac{\sqrt{r}}{m} \tag{4}$$

for every $(\gamma, \omega) \in \Gamma \times \Omega$. As a consequence, the sequence $\{\zeta_m\}$ of finitely supported primitive random fuzzy vectors converges to ξ uniformly.

In what follows, the sequence $\{\zeta_m\}$ of finitely supported primitive random fuzzy vectors is referred to as the discretization of ξ . The problem (II) is referred to as the original two-stage RFPR. By generating a sequence $\{\zeta_m\}$ of finitely supported primitive random fuzzy vectors according to the distribution of ξ , one can construct the corresponding approximation two-stage RFPR problem

$$\min_x \left\{ c^T x + \widehat{Q}_{m,E}(x) \mid x \in D \right\} \tag{5}$$

where $\widehat{Q}_{m,E}(x) = E_{\xi}[\widehat{Q}_m(x, \zeta_m)]$, and $\widehat{Q}(x, \zeta_m)$ is the optimal value of the following linear programming

$$\begin{cases} \min & q^T(\zeta_m)y \\ \text{subject to} & W(\zeta_m)y = h(\zeta_m) - T(\zeta_m)x \\ & y \geq 0. \end{cases}$$

The problem (5) is referred to as the approximating two-stage RFPR problem. The objective value of the approximating RFPR problem (5) provides an estimator for that of the original RFPR problem (1).

Theorem 1. *Consider the original RFPR problem (1), suppose the recourse matrix W is fixed, i.e., $W(\xi) \equiv W$, and $\xi = q$ or (h, T) is a continuous and bounded random fuzzy vector with an infinite support Ξ such that $Q(x, \xi)$ is not $-\infty$ for each feasible decision $x \in D$ and $\xi \in \Xi$. If the sequence $\{\zeta_m\}$ of finitely supported primitive random fuzzy vectors is the discretization of ξ , then for each $x \in D$, the objective value of the approximating RFPR problem (5) converges to that of the original RFPR problem (1), i.e.,*

$$\lim_{m \rightarrow \infty} \widehat{Q}_{m,E}(x) = Q_E(x)$$

where $\widehat{Q}_{m,E}(x) = E_{\zeta_m}[\widehat{Q}_m(x, \zeta_m)]$, and $Q_E(x) = E_{\xi}[Q(x, \xi)]$.

4 Hybrid Algorithm and Numerical Example

4.1 A Hybrid PSO Algorithm

If the demand is characterized by a continuous random fuzzy vector ξ , then (1) is inherently an infinite-dimensional optimization problem that cannot be solved directly by conventional optimization algorithms. To provide a general solution method to the two-stage RFPR problem, we will design a hybrid algorithm by integrating the PSO algorithm, NN and the approximation method. The PSO algorithm, originally developed by Kennedy and Eberhart (4), is a method for optimization on metaphor of social behavior of flocks of birds and/or schools of fish. Compared to other heuristic algorithm, the PSO algorithm has much better intelligent background, and the theoretical framework of PSO is very simple so that it can be performed easily. Recently the PSO algorithm has attracted much attention and been successfully applied in the fields of evolutionary computing, unconstrained continuous optimization problems and many others (5). In our proposed hybrid algorithm, the technique of the approximation method is used to compute the recourse function $Q_E(x)$, NN is trained to approximate the recourse function, and the PSO algorithm and the trained NN are integrated for solving the two-stage problem (5).

Training an NN: The evaluation of $Q_E(x)$ via approximation method is a time-consuming process since for each first-stage decision x and every realization $\zeta_{m,\gamma}(\omega)$ of ζ_m , we are required to solve the second-stage programming problem

via the simplex algorithm. To speed up the solution process, we desire to replace the recourse function $\mathcal{Q}_E(x)$ by an NN since a trained NN has the ability to approximate integrable functions. In this paper, we employ the fast BP algorithm to train a feedforward NN to approximate recourse function $\mathcal{Q}_E(x)$. We only consider the NN with input layer, one hidden layer and output layer connected in a feedforward way. Let $\{(x_i, q_i) \mid i = 1, 2, \dots, M\}$ be a set of input-output data generated by the approximation method. The training process is to find the best weight vector w so that the error function $Err(w) = \sum_{i=1}^M |F(x_i, w) - q_i|^2/2$ is minimized, where $F(x_i, w)$ is the output function of the NN, and q_i is the value of $\mathcal{Q}_E(x_i)$ evaluated by the approximation method.

Representation Structure: In the two-stage RFPR problem, we use a vector $x = (x_1, x_2, \dots, x_{n_1})^T$ as a particle to represent a decision variables.

Initialization: Initialize *pop_size* particles x^k for $k = 1, \dots, \text{pop_size}$ from the feasible region D .

Operations in PSO Algorithm: Suppose that the searching space is $2n_1$ -dimensional and there are *pop_size* particles from the colony. The position and the velocity of the k th particle can be represented as

$$x^k = (x_{k,1}, x_{k,2}, \dots, x_{k,n_1})^T, v^k = (v_{k,1}, v_{k,2}, \dots, v_{k,n_1})^T.$$

Each particle has its own best position (pbest) $p^k = (p_{k,1}, p_{k,2}, \dots, p_{k,n_1})^T$, which represents the personal smallest objective value so far at time t . The global best particle (gbest) of the colony is denoted by $p^g = (p_{g,1}, p_{g,2}, \dots, p_{g,n_1})^T$, which is the best particle found so far at time t in the colony.

Using the notations above, the new position of the k th particle is updated by

$$x^k(t+1) = x^k(t) + v^k(t+1) \quad (6)$$

while the new velocity of the k th particle is renewed by

$$v^k(t+1) = wv^k(t) + c_1r_1(p^k - x^k(t)) + c_2r_2(p^g - x^k(t)) \quad (7)$$

where $k = 1, 2, \dots, \text{pop_size}$; w is called the inertia coefficient; c_1 and c_2 are learning rates which are nonnegative constants, and r_1 and r_2 are two independent random numbers generated randomly in the unit interval $[0, 1]$.

Hybrid PSO Algorithm: To solve the proposed two-stage RFPR problem, we first employ the approximation method to generate a set of input-output data for the recourse function $\mathcal{Q}_E(x)$, then we use the data set to train an NN to approximate the recourse function $\mathcal{Q}_E(x)$. After the NN is well-trained, we integrate PSO and the trained NN to produce a hybrid algorithm. During the solution process, we use formula (6) to update the position of the k th particle, employ formula (7) to renew the velocity of the k th particle, and use the trained NN to calculate the objective values for all particles. We repeat the above process until a stopping criterion is satisfied.

The procedure of the proposed hybrid PSO algorithm for solving the two-stage RFPR problem is summarized as

- Step 1.** Generate a set of input-output data for the recourse function $Q_E(x)$ by the approximation method.
- Step 2.** Train an NN to approximate the recourse function $Q_E(x)$ by the generated input-output data.
- Step 3.** Initialize pop_size particles with random positions and velocities, and evaluate the objective values for all particles by the trained NN.
- Step 4.** Set $pbest$ of each particle and its objective value equal to its current position and objective value, and set $gbest$ and its objective value equal to the position and objective value of the best initial particle.
- Step 5.** Update the position and velocity of each particle according to formulas (6) and (7), respectively.
- Step 6.** Calculate the objective values for all particles by the trained NN.
- Step 7.** For each particle, compare the current objective value with that of its $pbest$. If the current objective value is smaller than that of $pbest$, then renew $pbest$ and its objective value with the current position and objective value.
- Step 8.** Find the best particle of the current swarm with the smallest objective value. If the objective value is smaller than that of $gbest$, then renew $gbest$ and its objective value with the position and objective value of the current best particle.
- Step 9.** Repeat the fifth to eighth steps for a given number of cycles.
- Step 10.** Return the $gbest$ and its objective value as the optimal solution and the optimal value.

4.2 One Numerical Example

To demonstrate the effectiveness of the designed hybrid PSO algorithm, consider the following two-stage RFPR problem

$$\left\{ \begin{array}{l} \min \quad 2x_1^2 + 2x_2^2 - 8x_3^2 + E_\xi[Q(x, \xi)] \\ \text{subject to } x_1 + x_2 + 2x_3 \leq 9 \\ \quad \quad \quad x_1 + x_2 - x_3 \leq 2 \\ \quad \quad \quad -x_1 + x_2 + x_3 \leq 4 \\ \quad \quad \quad x_1, x_2, x_3 \geq 0 \end{array} \right. \quad (8)$$

where $Q(x, \xi)$ is the optimal value of the following linear programming

$$\left\{ \begin{array}{l} \min \quad q_1(\xi)y_1 + q_2(\xi)y_2 + y_3 + q_4(\xi)y_4 + y_5 \\ \text{subject to } 2y_2 + y_3 - 4y_4 - 2y_5 = 30 - t_1(\xi)x_1 - x_2 + x_3 \\ \quad \quad \quad 5y_1 - 11y_2 + 7y_3 + 8y_4 - 10y_5 = 60 - 6x_1 - t_2(\xi)x_2 - 3x_3 \\ \quad \quad \quad -10y_1 - 8y_2 + 13y_3 + 5y_4 + 5y_5 = 80 - 12x_1 + 9x_2 - t_3(\xi)x_3 \\ \quad \quad \quad y_k \geq 0, k = 1, \dots, 5 \end{array} \right.$$

q_1, q_2, q_4 are uniform random fuzzy variables defined as

$$\begin{aligned} q_{1,\gamma} &\sim \mathcal{U}(5 - X_1(\gamma), 5 + X_1(\gamma)) \quad \text{with } X_1 \text{ triangular fuzzy variable } (0, 1, 2) \\ q_{2,\gamma} &\sim \mathcal{U}(3 - X_2(\gamma), 3 + X_2(\gamma)) \quad \text{with } X_2 \text{ triangular fuzzy variable } (0, 1, 2) \\ q_{4,\gamma} &\sim \mathcal{U}(2 - X_4(\gamma), 2 + X_4(\gamma)) \quad \text{with } X_4 \text{ triangular fuzzy variable } (0, 1/2, 1) \end{aligned}$$

t_1, t_2 and t_3 are the following normal random fuzzy variables

$$\begin{aligned} \mu_{t_1, \gamma}(r) &= \exp\left(-\left(\frac{r - Y_1(\gamma)}{0.5}\right)^2\right) \text{ with } Y_1 \text{ fuzzy variable } (-2, -1, 0) \\ \mu_{t_2, \gamma}(r) &= \exp\left(-\left(\frac{r - Y_2(\gamma)}{0.5}\right)^2\right) \text{ with } Y_2 \text{ fuzzy variable } (0, 1, 2) \\ \mu_{t_3, \gamma}(r) &= \exp\left(-\left(\frac{r - Y_3(\gamma)}{0.5}\right)^2\right) \text{ with } Y_3 \text{ fuzzy variable } (-10, -9, -8). \end{aligned}$$

In addition, the fuzzy variables involved in this problem are assumed to be mutually independent [13].

In order to solve (8), for each fixed first-stage decision variable x , we generate 1.5×10^6 sample points via the approximation method to calculate the recourse function $Q_E(x)$. Then for each sample point $\hat{\zeta}_{ij}$, we solve the second-stage programming via simplex algorithm and obtain the second-stage value $Q(x, \hat{\zeta}_{ij})$ for $i = 1, 2, \dots, 500$, and $j = 1, 2, \dots, 3000$. After that, the value of the recourse function $Q_E(x)$ at x can be computed by the approximation method.

By applying the approximation method described above, we first generate a set $\{(x_i, q_i) \mid i = 1, 2, \dots, 2000\}$ of input-output data for the recourse function $Q_E(x)$, and then use the data set to train an NN to approximate the recourse function $Q_E(x)$. After the NN is well trained, it is embedded into a PSO algorithm to produce a hybrid algorithm to search for the optimal solutions.

If we set the parameters in the implementation of the PSO algorithm as follows: the inertia coefficient w decreases linearly from 0.9 to 0.4; the learning rates $c_1 = c_2 = 2$, and the population size is 100, then a run of the hybrid PSO algorithm with 400 generations gives the following optimal solution

$$(x_1^*, x_2^*, x_3^*) = (0.333365, 0.000000, 4.333317)$$

whose objective value is -99.846520 .

5 Conclusion

This paper has developed a new class of two-stage RFPR problem. Since the uncertain parameters in the two-stage RFPR problem are often modeled as random fuzzy variables with infinite supports, it is inherently an infinite-dimensional optimization problem that can rarely be solved directly by conventional optimization algorithms. To overcome this difficulty, this paper has presented an approximation method to the original two-stage RFPR problem, and established a convergence relation between the original two-stage RFPR problem and its approximating problem. Furthermore, we have designed a hybrid PSO algorithm by integrating the approximation method, NN and the PSO algorithm to solve the proposed RFPR problem. The effectiveness of the designed algorithm was illustrated via one numerical example.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No.70571021), the Program for One Hundred Excellent and Innovative Talents in Colleges and Universities of Hebei Province, and the Natural Science Foundation of Hebei Province (No.A2008000563).

References

1. Dantzig, G.B.: Linear Programming Under Uncertainty. *Management Science* 1, 197–206 (1955)
2. Feng, X., Liu, Y.K.: Measurability Criteria for Fuzzy Random Vectors. *Fuzzy Optim. Decision Making* 5, 245–253 (1955)
3. Guo, R., Zhao, R., Guo, D., Dunne, T.: Random Fuzzy Variable Modeling on Repairable System. *Journal of Uncertain Systems* 1, 222–234 (2007)
4. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: *Proc. of the IEEE International Conference on Neural Networks*, Piscataway, NJ, pp. 1942–1948 (1995)
5. Kennedy, J., Eberhart, R.C., Shi, Y.: *Swarm Intelligence*. Morgan Kaufmann Publishers, San Francisco (2001)
6. Liu, B.: A Survey of Credibility Theory. *Fuzzy Optimization and Decision Making* 5, 387–408 (2006)
7. Liu, B.: *Theory and Practice of Uncertain Programming*. Physica-Verlag, Heidelberg (2002)
8. Liu, B.: *Uncertainty Theory*. Springer, Berlin (2004)
9. Liu, B., Liu, Y.K.: Expected Value of Fuzzy Variable and Fuzzy Expected Value Models. *IEEE Trans. Fuzzy Syst.* 10, 445–450 (2002)
10. Liu, Y.K.: Fuzzy Programming with Recourse. *Int. J. Uncertainty Fuzziness Knowledge Based Syst.* 13, 381–413 (2005)
11. Liu, Y.K.: Convergent Results About the Use of Fuzzy Simulation in Fuzzy Optimization Problems. *IEEE Trans. Fuzzy Syst.* 14, 295–304 (2006)
12. Liu, Y.K., Dai, X.: The Convergence Modes in Random Fuzzy Theory. *Thai Journal of Mathematics* 6 (2008)
13. Liu, Y.K., Gao, J.: The Independence of Fuzzy Variables with Applications to Fuzzy Random Optimization. *Int. J. Uncertainty Fuzziness Knowledge Based Syst.* 15, 1–20 (2007)
14. Liu, Y.K., Liu, B.: Random Fuzzy Programming with Chance Measures Defined by Fuzzy Integrals. *Math. Compu. Model* 36, 509–524 (2002)
15. Liu, Y.K., Liu, B.: Expected Value Operator of Random Fuzzy Variable and Random Fuzzy Expected Value Models. *Int. J. Uncertainty Fuzziness Knowledge Based Syst.* 11, 195–215 (2003)
16. Liu, Y.K., Liu, B., Chen, Y.: The Infinite Dimensional Product Possibility Space and Its Applications. In: Huang, D.-S., Li, K., Irwin, G.W. (eds.) *ICIC 2006*. LNCS (LNAI), vol. 4114, pp. 984–989. Springer, Heidelberg (2006)
17. Liu, Y.K., Liu, Z.Q., Liu, Y.: Fuzzy Optimization Problems with Critical Value-at-Risk Criteria. In: Liu, D., Fei, S., Hou, Z., Zhang, H., Sun, C. (eds.) *ISSN 2007*. LNCS, vol. 4492, pp. 267–274. Springer, Heidelberg (2007)
18. Liu, Y.K., Zhu, X.: Capacitated Fuzzy Two-Stage Location-Allocation Problem. *Int. J. Innovative Computing, Information and Control* 3, 987–999 (2007)

A Fuzzy Logic Approach to Test Statistical Hypothesis on Means

Juan Carlos Figueroa García¹ and Jose Jairo Soriano Mendez²

¹ Universidad Distrital Francisco José de Caldas, Bogotá - Colombia*
jcfigueroag@udistrital.edu.co

² Universidad Distrital Francisco José de Caldas, Bogotá - Colombia
josoriano@udistrital.edu.co

Abstract. This paper presents a generalized Type-1 Fuzzy Logic engine to test statistical hypothesis on means by using standardized data samples to simplify the rule base. This inference engine attempts to test hypothesis on imprecise means, being an alternative to reject or accept the hypothesis via a fulfillment degree. To do so, an application example is provided and compared against classical tests to verify their results.

1 Introduction and Motivation

Recently, the statistical theory is focused in the analysis of different situations related with data quality. This statistical field known as *Robust Statistics* considers some problems as missing data, biased estimations, atypical data and low quality measurements. For further information see [1], [2] and [3].

A new field on statistical analysis is focused on the usage of fuzzy theory to handle imprecise data. James J. Buckley in [4] and [5] describes new probability concepts based on Type-1 fuzzy sets called “*Fuzzy Probabilities*” and therefore fuzzy test statistics. A. Mohammadpuor & A. Mahammad.Djafari in [6] propose a fuzzy test by using fuzzy relations and Bayesian concepts and B. F. Arnold in [7] proposes a similar work based on interval numbers.

In this way a Type-1 Fuzzy Logic (T1 FLS) engine is proposed to handle low quality and unbiased samples assuming noise, damage or malfunction of the measurement devices and related issues.

2 Hypothesis Testing

A Statistical Hypothesis is an assertion or conjecture about the distribution of one or more random variables. A test of a statistical hypothesis (H) is a rule or procedure for deciding whether to reject (H) or not.

In many hypothesis-testing problems two hypothesis are discussed: The first, the hypothesis being tested, is called the *Null Hypothesis*, denoted by H_0 , and

* Both authors are members of the *Laboratory for Automation, Microelectronics and Computational Intelligence (LAMIC)*, Bogotá - Colombia.

the second is called the *Alternative Hypothesis*, denoted by H_a . The natural supposition is that if H_0 is false, then H_a is true.

Different H_0 can be outlined: Differences between μ and μ_0 , between μ_1 and μ_2 or among multiple means $\{\mu_1, \mu_2, \dots, \mu_n\}$. Their definitions are shown next:

$$H_0 : \mu = \mu_0 \tag{1}$$

$$H_a : \mu \neq \mu_0; \quad H_a : \mu < \mu_0 \quad \text{or} \quad H_a : \mu > \mu_0$$

$$H_0 : \mu_1 = \mu_2 \tag{2}$$

$$H_a : \mu_1 \neq \mu_2; \quad H_a : \mu_1 < \mu_2 \quad \text{or} \quad H_a : \mu_1 > \mu_2$$

$$H_0 : \mu_1 = \mu_2 = \dots = \mu_n \tag{3}$$

$H_a : \text{At least one mean is different from others.}$

Here, H_0 is tested with any test statistic with a pre-defined α confidence level (Usually $\alpha = 0.05$). Usually, the analyst verifies his suppositions about (1) and (2) by using test statistics as the z or t – student statistics, assuming them as the correct choice, based on asymptotical assumptions.

The multiple means case (3) is widely treated by using the ANOVA methodology due to their complexity. For further information see [2], [3] and [8].

2.1 Decision Making

A crisp bilateral hypothesis test is a simple process that either rejects or accepts H_0 by using a test statistic based on a known *pdf*¹. The usual reasoning is to Accept H_0 if the sample statistic is inside a *Confidence Interval*, that is:

$$\bar{x} \in [g_1(\mu_0); g_2(\mu_0)] \tag{4}$$

Where $g_1(\mu_0)$ and $g_2(\mu_0)$ are functions of μ_0 and $P(g_1(\mu_0) < \mu < g_2(\mu_0)) = 1 - \alpha$.

The classical approach finds the probability of satisfy H_0 . If the sample statistic falls on a extreme value then H_0 is rejected, otherwise it “*Does not have any statistical evidence to reject H_0* ” and H_a is accepted. Here, the principal result is a boolean decision about H_0 : Reject it or not.

2.2 Uncertainty Sources in Hypothesis Testing

An important discussion around sources of uncertainty on Type-1 and Type-2 FLS have been made by Mendel in [9], Klir in [10] and [11] and Melgarejo in [12], and [13]. Thus, the following sources of uncertainty have been identified:

Data quality: The measurement device would have problems or errors on its functioning and finally it generates bias from their main information.

Ambiguity (Confusion on evidence): The data sample could contain undesirable information (Noise), leading to errors on their estimates.

Ambiguity (Dissonance on evidence): When samples has no Normal behavior, their multivariate *pdf* could leads to a misspecification of the test.

These uncertainty sources have not been considered by classical approaches, but a FLS can work with. To do so, a non-parametric fuzzy test is proposed.

¹ Probability Distribution Function.

2.3 Fuzzy Random Vectors

In this paper, the definition of fuzzy randomness given by Yian-Kui Liu & Baoding Liu in [14] is used.

Definition 1. Let (Ω, Σ, P) be a probability space. A *f.r.v.* is a mapping $\xi : \Omega \rightarrow \mathfrak{F}_v$ such that for any closed subset C of \mathbb{R} .

$$\xi(C)(\omega) = Pos\{\xi(\omega) \in C\} = \sup_{x \in C} \mu_{\xi(\omega)}(x) \tag{5}$$

Is a measurable function of ω , where $\mu_{\xi(\omega)}$ is the distribution function of fuzzy variable $\xi(\omega)$.

According to above, a *f.r.v.* is defined both in probability and possibility spaces, by means of a membership function $\mu_{\xi(\omega)}$ in the ω fuzzy space.

3 Type-1 Inference Engine to Test Hypothesis on Means

First, a statistical transformation is used to standardize the sample mean namely $\{\bar{x}_i\}$. For known and unknown variances, the z and t transformations are:

$$z_i = \frac{\bar{x}_i - \mu_0}{\delta_i / \sqrt{n_i}} \tag{6}$$

$$t_i = \frac{\bar{x}_i - \mu_0}{s_i / \sqrt{n_i}} \tag{7}$$

\bar{x}_i, δ_i, s_i and n_i are the mean, population and sample variance of the i_{th} sample.

By using *f.r.v.* and (5), both imprecision and randomness are treated by means a T1 FLS whose methodology is presented next.

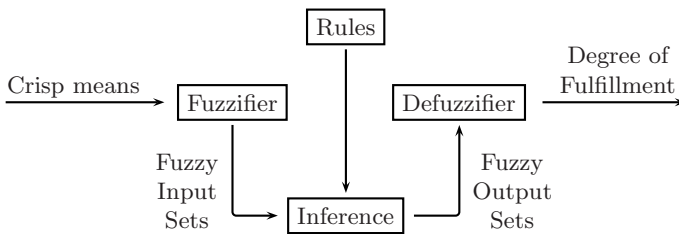


Fig. 1. Type-1 FLS design methodology

3.1 General Methodology

General facts about the proposed fuzzy test are:

- The i_{th} group mean must be standardized by using (6) or (7), eliminating the effect of its variance and simplifying the rule base.
- All designed tests are based on bilateral differences on means.
- The defined *f.r.v.* does not obey to any probability function.

The methodological process in order to design a FLS is presented next.

3.2 Fuzzy Input Sets

First, some considerations about properties of the samples can be given.

Definition 2. *The fuzzy space of a z or t transformation is given by all values defined in the map $\xi : \Omega \rightarrow \mathfrak{F}_v$ such that for any closed subset \mathcal{C} of \mathbb{R} .*

Lemma 1. *If the i_{th} mean is equal to their reference (μ_0) then $z = 0$.*

Proof. Here, $z_i(\omega) = (\bar{x}_i(\omega) - \mu_0)/(\delta_i/\sqrt{n_i})$, now consider any i_{th} mean where $\bar{x}_i(\omega) = \mu_0$, then $z_i(\omega) = (\mu_0 - \mu_0)/(\delta_i/\sqrt{n_i})$ and finally $z_i(\omega) = 0$. ω is the sample space, $i \in \mathbb{N}$, $z_i(\omega) \in \mathcal{C}$ and $\bar{x}_i \in \mathbb{R}$. For an unknown variance case we use [7] and their proof is similar to the above result. \square

In this paper, we use Gaussian membership functions $G(a_0, b_i)$ for all input sets (See [8]). A central set “Close to μ_0 ” is characterized by $G(a_0, b_i)$, where $a_i = z_i$, $a_0 = 0$ and b_i which is a free parameter of exigency of the test.

$$\mu_c(a_i; a_0, b_i) = \exp \frac{-(a_i - a_0)^2}{2b_i^2} \tag{8}$$

Thus, [8] is the fuzzy distribution function of the behavior of z or $t \rightarrow \mu_c$, defined on the closed set $C(z, t) : (-\infty, \infty)$, See [5].

Remark 1. *Recall that Yian-Kui Liu and Baoding Liu in [14] shows that the expected value of a f.r.v. with gaussian membership function is exactly a_0 , in this case is zero (0), and also is the central tendency of the z or t transformation.*

By extension it is easy to show that $E[z(\omega)] = 0$, $\sum_{j=1}^n (z_j(\omega) - E[z(\omega)]) = 0$ and $E[t(\omega)] = 0$, $\sum_{j=1}^n (t_j(\omega) - E[t(\omega)]) = 0$ for any random sample.

Three input sets: *Smaller* (s), *Close* (c) and *Bigger* (b) than μ_0 are defined as:

$$\text{“Smaller than } \mu_0 \text{” } (\mu_s) : 1 - \exp^{-\frac{1}{2} \left(\frac{\bar{x} - \mu_0}{\delta/\sqrt{n}} \right)^2} \quad \text{for } z < 0 \tag{9}$$

$$\text{“Close than } \mu_0 \text{” } (\mu_c) : \exp^{-\frac{1}{2} \left(\frac{\bar{x} - \mu_0}{\delta/\sqrt{n}} \right)^2} \quad \forall z \tag{10}$$

$$\text{“Bigger than } \mu_0 \text{” } (\mu_b) : 1 - \exp^{-\frac{1}{2} \left(\frac{\bar{x} - \mu_0}{\delta/\sqrt{n}} \right)^2} \quad \text{for } z > 0 \tag{11}$$

Here, $a_i = \bar{x}$, $a_0 = \mu_0$ and $b_i = \delta/\sqrt{n}$. μ_s, μ_c and μ_b are shown in the Figure 2.

The definition of all fuzzy sets is a flexible aspects which depends on the opinion and perception of the experts about \bar{x} regarding μ and z .

Remark 2 (Equally on variances supposition:). *Additionally, the analyst can suppose that all groups have either equal variances or not. In the case where the analyst does not assume that, the proper z and t transformations are:*

$$z_i(\omega) = \frac{\bar{x}_i(\omega) - \mu_0}{\delta_i/\sqrt{n_i}} \tag{12}$$

$$t_i(\omega) = \frac{\bar{x}_i(\omega) - \mu_0}{s_i/\sqrt{n_i}} \tag{13}$$

For a supposition of equality on variances, their proper expressions are:

$$z_i(\omega) = \frac{\bar{x}_i(\omega) - \mu_0}{\delta_s / \sqrt{n_i}} \tag{14}$$

$$t_i(\omega) = \frac{\bar{x}_i(\omega) - \mu_0}{s_s / \sqrt{n_i}} \tag{15}$$

Where δ_s^2 is the population variance and s_s^2 is the sample variance defined as:

$$s_s^2 = \frac{1}{N-1} \sum_{i=1}^a \sum_{j=1}^{n_i} (x_{ij} - \bar{x}_{..})^2 \tag{16}$$

3.3 Fuzzy Output Sets

Three fuzzy output sets are defined as Low (*l*), Medium (*m*) and High (*h*) fulfillment of H_0 , whose mathematical expressions are given next:

$$\text{“Low Fulfillment of } \mu_0 \text{”} : \mu_l = \begin{cases} 1 - 2 \left(\frac{f+0.2}{0.7} \right)^2 & -0.2 \leq f \leq 0.15 \\ 2 \left(0.5 - \frac{f}{0.7} \right)^2 & 0.15 \leq f \leq 0.5 \end{cases} \tag{17}$$

$$\text{“Medium Fulfillment of } \mu_0 \text{”} : \mu_m = \exp^{-\left(\frac{f-0.5}{2\delta}\right)^2} \forall f \tag{18}$$

$$\text{“High Fulfillment of } \mu_0 \text{”} : \mu_h = \begin{cases} 2 \left(\frac{f-0.5}{0.7} \right)^2 & 0.5 \leq f \leq 0.85 \\ 1 - 2 \left(0.5 - \frac{f}{0.7} \right)^2 & 0.85 \leq f \leq 1.2 \end{cases} \tag{19}$$

Where f is the final output of the FLS and δ is a spread parameter which offers the choice of either to increase or reduce the exigency of the test. Note that $f \in [-0.2; 1.2]$ since the defuzzification process does not provide *zero* or *one* as extreme fulfillments of H_0 when the FLS is constructed with other values. A graphical representation is displayed next:

The output sets has domain $\mathbb{R} \in [0, 1]$, where zero means a very low fulfillment of H_0 and one is a complete fulfillment, making easy their interpretation.

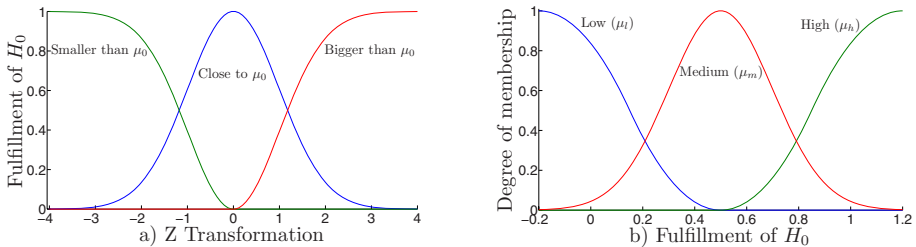


Fig. 2. a) Input sets and b) Output sets

3.4 Rule Base

The classical statistical hypothesis only uses one rule to verify H_0 , based on the asymptotic properties (Usually a Normal behavior) of the sample:

R_1 : If $\bar{x} \in [g_1(\mu_0); g_2(\mu_0)]$ then H_0 is Accepted, otherwise H_0 is Rejected.

We represent the knowledge of the experts about H_0 by means of a rule-base. A general way to design bilateral tests is presented next.

Rule Base for one mean test:

R_1 : if z is not c , then output is l .

This FLS obtains a similar result than the approach developed by Buckley in [4] and [5]. Their response surface is depicted in the Figure 3.

Rule Base for multiple means test: A multiple means test tries to verify $H_0 : \mu_1 = \mu_2 = \dots = \mu_n$ by using the “Mean of complete data” shown next:

$$\bar{y}_{..} = \sum_{i=1}^a \sum_{j=1}^n \frac{y_{ij}}{N} \tag{20}$$

If all means are similar among them, then $\bar{y}_{..}$ is close them, but if at least one differs from the rest, then $\bar{y}_{..}$ is biased regarding them. By using a Pareto reasoning, the following propositions are given.

Proposition 1. *Be H_0 a fuzzy test of multiple means, then $\{H_0 : \mu_1 = \mu_2 = \dots = \mu_n\}$ is true iff all means are close to (20), having a high fulfillment of H_0 .*

Proposition 2. *Be H_0 a fuzzy test of multiple means, then $\{H_0 : \mu_1 = \mu_2 = \dots = \mu_n\}$ is not true iff one or more means differ from (20). If more than 30% of means differs from (20), then the fulfillment of H_0 is low and if at least one or less than 30% of means differ from (20), then the fulfillment of H_0 is medium.*

By this way, the following rules are obtained for the two means case:

- R_1 : if z_1 and z_2 are c , then output is h .
- R_2 : if z_1 and z_2 are not c , then output is l .
- R_3 : if z_1 is c and z_2 is not c , then output is l .
- R_4 : if z_1 is not c and z_2 is c , then output is l .

For three means, the following rule base is presented:

- R_1 : if z_1, z_2 and z_3 are c , then output is h .
- R_2 : if z_1, z_2 and z_3 are not c , then output is l .
- R_3 : if z_1 is not c and z_2 is c and z_3 is c , then output is m .
- R_4 : if z_1 is c and z_2 is not c and z_3 is c , then output is m .
- R_5 : if z_1 is c and z_2 is c and z_3 is not c , then output is m .
- R_6 : if z_1 is not c and z_2 is not c and z_3 is c , then output is l .
- R_7 : if z_1 is not c and z_2 is c and z_3 is not c , then output is l .
- R_8 : if z_1 is c and z_2 is not c and z_3 is not c , then output is l .

If the sample has more than 3 data groups, a similar reasoning is useful to extract rules from samples. Their response surfaces are shown in the Figure 3).

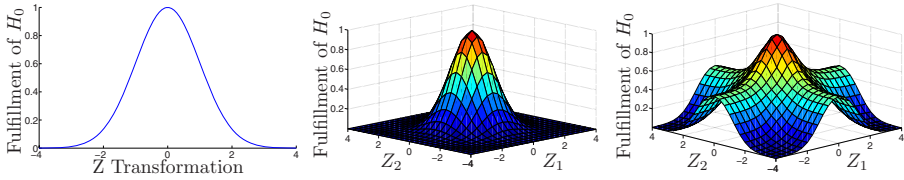


Fig. 3. Response Surface for One, Two and Three Means FLS

3.5 Implications and Defuzzification Process

Different operations on T1 FS was implemented, but the better results of the T1 FLS are obtained by the following operators: *And* operator: *Product*, Implications: *Product*, Aggregation operator: *Sum* and Defuzzification: *Centroid*.

Definition 3. *The output value after defuzzification is defined as the P_s – value named $\{P_s v\}$ of fulfillment of H_0 . It is a defuzzified measure of the overall fulfillment of all means regarding H_0 . Formally:*

$$P_s v = \frac{\int \mu_f(f) f \, df}{\int \mu_f(f) df} \tag{21}$$

This $P_s v$ defines if all data-groups either should be reviewed or not to detect differences among them.

4 Decision Making Process

As in the classical hypothesis testing process, the analyst should make a decision about H_0 based on the probability of make a Type-I error. In the proposed approach, he uses a fulfillment degree regarding H_0 , but it does not reject it.

By using the Definition 3, a Pareto-based reasoning to contrast H_0 by means of $\{P_s v\}$ is proposed next:

- R_1 : If the obtained $P_s v$ is greater than 70%, then H_0 is completely true.
- R_2 : If the obtained $P_s v$ is enclosed into the interval [30%–70%], then H_0 is true but data groups could contain some small differences among them, caused by probabilistic or possibilistic fluctuations inherent to samples.
- R_3 : If the $P_s v$ is smaller than 30%, then H_0 could be not true and at least one mean is different from others. An exhaustive test for all means is recommended to identify differences.

Remark 3. *The fuzzy test does not attempt to reject or accept H_0 , only for a low $P_s v$ degree it suggests to contrast all samples by using other techniques, but for other $P_s v$ degrees it does not reject H_0 . For a medium $P_s v$ degree, it suggests the existence of small differences, and for a high $P_s v$ degree, it accepts H_0 .*

Finally, let us point out the soft nature of the proposed test which attempts to detect differences among samples under ambiguity, imprecision and uncertainty conditions. Only for low fulfillments of H_0 , the test recommends an exhaustive comparison, but in other cases, H_0 is Accepted.

5 Application Example

An experiment is implemented to contrast the equality on means hypothesis $\{H_0 : \mu_1 = \mu_2 = \mu_3\}$ among three groups, obtaining the following results:

Table 1. Resultant experiment report

Group	Size (n)	Sum	Mean	Variance
1	30	602.952	20.098	28.567
2	30	522.690	17.423	15.403
3	30	601.666	20.056	24.736

The ANOVA test gives $F = 3.076$ and a $p - value$ of 0.051. By using these results the analyst can not reject H_0 and he would concludes that all means are equal. On the other hand, the $p - value$ is very close to 0.05, generating doubts around their significance. Additionally, the analyst points out measurement problems in the 2nd group, identifying a source of uncertainty.

By using the fuzzy proposal and (12), the analyst can reject H_0 accepting H_a because the obtained P_{sv} is less than 0.3, it means that data groups have different means. Their results are shown in the Table 2

Table 2. Resultant Fuzzy Test on means

Group	Sum	Mean	Variance	t_i	P_{sv}
1	602.952	20.098	28.567	0.929	0.267
2	522.190	17.406	15.296	-2.469	
3	601.666	20.056	24.736	0.951	

Due to the ANOVA test accepts H_0 and the fuzzy approach identify differences among means, four comparison tests are applied to contrast H_0 . According to S.G. Carmer and M.R. Swanson (See (15)), the Duncan and Gabriel tests are applied to find differences among means.

In the Table 3, each test contrasts the significance of the grouped samples. Only significant groups are shown. A comparison fuzzy test is done to find homogeneous

Table 3. Comparison Tests for $\alpha = 0.05$

Test	Significance of the Subset.		
	1	2-3	1-2-3
Duncan	—	0.972	—
Waller-Duncan	—	0.846	—
Gabriel	—	—	0.096
Fuzzy-Test	—	0.999	—

groups. To do so, the two means rule-base is used for, finding that the the 2^{nd} sample differs from the 1^{st} and 3^{rd} samples².

So, H_0 is Rejected since the 2^{nd} sample is different from the rest.

6 Concluding Remarks

Some conclusions and recommendations of the study can be suggested:

- By using the proposed fuzzy test the analyst can get more information about imprecise samples and their statistical properties. A soft computing comparison is proposed and implemented with successful results.
- The classical statistical tests only reject or not H_0 , but the proposed fuzzy approach does not only reject H_0 , it suggests the use of other tests to identify differences among means, whenever any uncertainty source exists.
- The proposed T1 FLS detects ambiguities on classical tests, being a complementary technique to contrast imprecise means.
- This proposal is a non-parametric option for statistical analysis, easily to be implemented in MatLab®, Excell, OpenOffice or any database platform.

Finally, it is important to emphasize that the proposed test does not replace any other test. Its nature only turns it into an information criterion.

References

1. Huber, P.: Robust Statistics. John Wiley and Sons, New York (1981)
2. Mood, A.M., Graybill, F.A., Boes, D.C.: Introduction to the Theory of Statistics. McGraw-Hill Book Company, New York (1974)
3. Wilks, A.: Mathematical Statistics. John Wiley and Sons, Chichester (1962)
4. Buckley, J.J.: Fuzzy Probability and Statistics. Springer, Heidelberg (2004)
5. Buckley, J.J.: Fuzzy statistics: hypothesis testing. Soft Computing - A Fusion of Foundations, Methodologies and Applications 9(7), 512–518 (2005)
6. Mohammadpuor, A., Mahammad.Djafari, A.: On Classical, Bayesian and Fuzzy Hypothesis Testing. Metrika 17(2), 348–354 (2004)
7. Arnold, B.: An approach to fuzzy hypothesis testing. Metrika 44(2), 119–126 (1996)
8. Montgomery, D.: Analysis and Design of Experiments. John Wiley and Sons, Chichester (2006)
9. Mendel, J.: Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice Hall, Englewood Cliffs (1994)
10. Klir, G., Folger, T.: Fuzzy Sets, Uncertainty and Information. Prentice Hall, Englewood Cliffs (1992)
11. Klir, G., Yuan, B.: Fuzzy Sets and Fuzzy Logic: Theory and Applications. Prentice Hall, Englewood Cliffs (1995)

² The fuzzy test groups the 1^{st} and 2^{nd} samples with a $P_s v = 0.0865$ and the 1^{st} and 3^{rd} samples with a $P_s v = 0.0648$.

12. Melgarejo, M., Garcia, R., Peña, C.: Pro-two: A hardware based platform for real time type-2 fuzzy inference. In: IEEE Intl. Conf. on Fuzzy Systems, pp. 977–982 (2004)
13. Melgarejo, M.A.: Implementing Interval Type-2 Fuzzy processors. IEEE Computational Intelligence Magazine 2(1), 63–71 (2007)
14. Liu, Y.K., Liu, B.: Fuzzy random variables: A scalar expected value operator. Fuzzy Optimization and Decision Making 2(2), 143–160 (2003)
15. Carmer, S., Swanson, M.: Evaluation of ten pairwise multiple comparison procedures by monte carlo methods. Journal of the ASA 68(314), 66–74

A Fuzzy Logic-Based Approach to Two-Dimensional Warranty System

SangHyun Lee¹, DongSu Lee¹, ChulSu Park¹, JaeHee Lee¹, SeungBeom Park¹,
KyungIl Moon², and ByungKi Kim¹

¹ Department of Computer Engineering, Chonnam National University, Korea

² Department of Computer Engineering, Honam University, Korea

Leesang64@gmail.com, tnsqnfnn@naver.com, av3616@chonnam.ac.kr,
jhlee@chonnam.ac.kr, puru21@msn.com, kimoona@honam.ac.kr,
bgkim@chonnam.ac.kr

Abstract. Much of the literature on warranty analysis considers failure models which are indexed by a single variable, such as age or usage. However, there are situations where several characteristics are used together as criteria for judging the eligibility of a failed product. The warranty analysis characterized by a region in a two-dimensional plane with one axis representing age and the other axis representing usage is known as a “two-dimensional” warranty plan. A classical warranty plan requires crisp data obtained from strictly controlled reliability tests. However, in a real situation these requirements might not be fulfilled. In an extreme case, the warranty claims data come from users whose reports are expressed in a vague way. It might be caused by subjective and imprecise perception of failures by a user, by imprecise records of warranty data, or by imprecise records of the rate of usage. This paper suggests different tools appropriate for modeling a two-dimensional warranty plan, and a suitable fuzzy method to handle vague data.

Keywords: Two-dimensional warranty; Usage accumulation ratio; Warranty claims recurrence; Warranty costs analysis; Fuzzy logic.

1 Introduction

The most important factors are the age of the product and the effects of the manufacturing characteristics, time of manufacture and the operating seasons or environments. Age is measured by calendar time in terms of years, while usage is measured by real operating time in terms of such items as mileage and number of copies.

Moskowitz and Chun (1994) suggested a Poisson regression model for a two-dimensional warranty plan [3]. They assumed that the number of events under two-dimensional warranty policies is distributed as a Poisson. In a two-dimensional warranty policy, however, it seems to be unreasonable to restrict the type of failure rate function to strictly age and usage. Singpurwalla and Wilson (1998) proposed an approach for developing probabilistic models in a reliability setting indexed by two variables, i.e. age and an age-specific quantity such as amount of use [6]. However, it is not reasonable to use these variables in an additive hazard model.

Kim and Rao (2000) considered a two-attribute warranty plan for non-repairable items, to deal with the expected warranty cost analysis based on a bivariate exponential function [1]. Majeske (2003) proposed a general mixture model framework for automobile warranty data [2]. Pal and Murthy (2003) presented an application of Gumbel’s bivariate exponential function in estimation of warranty cost under a two-attribute warranty policy [4]. Although the exponential function is still frequently used because of two important features (namely, its parameter is easily estimated, and for age described by a probability distribution with increasing hazards, it gives a conservative approximation for the mean age), this assumption is very restrictive. Rai and Singh (2003) discussed a method to estimate hazard rate from incomplete and unclear warranty data [5]. However, sometimes we face situations when the number of observed failures is also vague.

Classical estimators require precise data obtained from strictly controlled reliability tests. In such a case, a failure should be precisely defined, and all tested items should be continuously monitored. However, in a real situation these requirements might not be fulfilled. We need different tools appropriate for modeling warranty data, and suitable reasoning methodology to handle these data as well. In this paper, we suggest another generalization of the classical two-attribute warranty plan. We consider not only fuzzy lifetimes but also situations in which the usage is fuzzy as well.

2 Classical Approach

2.1 Usage Accumulation Data and Repeat Claims

Let M_i denote time (in months from the sale of the automobile) and K_i denote mileage for the i -th ($i = 1, 2, \dots, N$) automobile in a population of same type of vehicles. Let $n_i(m)$ denote mileage accumulation of the i -th automobile. Let $F_i(t)$ and $f_i(t)$ be the distribution function (df) and probability density function (pdf), respectively, for M_i . Estimating the parameters of pdf and df is a crucial step in the analysis. A warranty database usually contains mileage accumulation data for only those cars that fail within the warranty period. Repeat claims could be the result of either a new failure or difficulty in root cause elimination during the previous repair. The expected number of total claims can be obtained by combining estimates of repeat claims with the estimates for the first claims. Repeat claims as a proportion of the first claims can be estimated using following formula.

$$\hat{p}_{rc, m / k_0} = \frac{n(m) - n_f(m)}{n_f(m)}$$

- 1) $\hat{p}_{rc, m / k_0}$: estimate of repeat claims as a proportion of the first claims at m month-in-service and K_0 mileage warranty limit.
- 2) $n(m)$: total number of claims up to m month-in-service value
- 3) $n_f(m)$: number of first claims up to m_0 month-in-service or number of cars with at least one claim up to m_0 month-in-service.

When increments of 5 or 10 month-in-service are used for arriving at $p_{rc,m/k_0}$ values, the curve fitted to the data points can be used to arrive at intermediate $p_{rc,m/k_0}$ values.

Using $\hat{p}_{rc,m/k_0}$ and parameters of $G(\alpha)$, $\hat{p}_{rc,m/ unlimited}$ is estimated as:

$$\hat{p}_{rc,m/ unlimited} = \frac{x - v}{p[\alpha_i \leq \frac{K_0}{M_0} | m]}$$

2.2 Cost of Warranty Claims

The cost per unit sold mainly includes time taken by the service technician to successfully repair the vehicle and the material cost. The absolute value of cost per repair may also depend on the nature of failure mode. On the other hand, the cost to repair a vehicle experiencing an engine failure may run into several thousand dollars. Obtaining warranty cost per unit sold for a warranty system requires estimates of expected C/100 and the cost per repair at different month-in-service values. In this section, a method for arriving at warranty cost per unit sold using estimates for C/100 and incremental cost per repair is discussed. A curve fitted to $c_r(m)$ versus t captures the changes in warranty cost per repair as a function of month-in-service. A curve fitted to $c_r(m)$ versus m captures the changes in warranty cost per repair as a function of month-in-service. Some of the major factors are nature and time of the occurrence of a failure mode, mileage accumulation rates for a population of cars, and the design actions such as design for serviceability. The parametric model fitted to the data gives information regarding nature of the failure mode under study. The cumulative cost per repair denoted by:

$$C_{cost} = \frac{\sum_{t=1}^{M_0} [M_{total}(t / K_0) - M_{total}((t - 1) / K_0)] \times c_r(t)}{M_{total}(\frac{m}{K_0})}$$

A population of cars with relatively high mileage accumulation rates may not experience time-related failure modes within the warranty period due to early crossing of the mileage limit.

3 Fuzzy Reasoning of Two-Dimensional Warranty

Fuzzy logic was originally introduced by Zadeh as a mathematical way to represent vagueness in everyday life. The proposed overall procedure for approximating the impact of corrective actions on two-dimensional warranty is shown in Fig. 1, which consists of four components, namely, fuzzy rule base, fuzzy inference process, fuzzification process, and defuzzification process.

The basic unit of any fuzzy system is the fuzzy rule base. All other components of the fuzzy logic system are used to implement these rules in a reasonable and efficient manner.

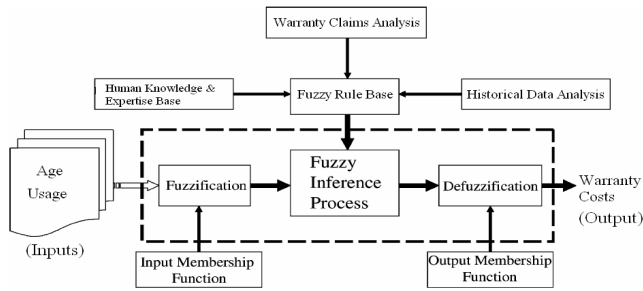


Fig. 1. An overview of fuzzy logic system for a two-dimensional warranty

Fuzzy rules are usually formulated as IF-THEN statements, with one or more antecedents connected to a consequent via operators like AND, OR, etc.

IF (Antecedent₁) OP (Antecedent₂) ... OP (Antecedent_n) THEN (Consequent) (*w*)

where *n* is an integer, OP stands for operators like AND, OR, etc., and *w* represents a weight value indicating the importance of a rule. Now, imagine a fuzzy rule where 2 antecedents apply to the same consequent (*n* = 2). Further, let Antecedent₁ be activated to a degree of 0.8, and Antecedent₂ to a degree of 0.7. The weight value is usually 1.0, and OP is usually an OR operator defined as: Consequent = max[Antecedent₁, Antecedent₂]. In this situation, Consequent would be activated to a degree of max[0.8, 0.7] = 0.8. There is nothing extraordinary here. The process described is a standard process. Now imagine a more complex scenario where 5 antecedents apply to a Consequent (*n* = 5). A possible scenario may look like: Consequent = max[0.7, 0.8, 0.6, 0.5, 0.9] = 0.9. In this situation, we probably are less confident in the applicability and usefulness of the rule. Furthermore, as there are many different weights indicating the importance of the rule, an aggregation of the consequents across the rules is probably less confident. The formulation of rules showing such complexity, however, might be common in some domains. To approach the problem, this paper presents a method that aims to include each activated rule antecedent more actively in the reasoning process of the fuzzy rule base.

To explain the method, we describe a simple fuzzy rule system where two antecedents (Age, Usage) relate to an output (Warranty Cost). The following two rules shall be included in the fuzzy rule system:

IF (Age) IS (*normal*) OR (Usage) IS (*normal*) THEN (Warranty Cost) is (*normal*)
 IF (Age) IS (*low*) OR (Usage) IS (*low*) THEN (Warranty Cost) is (*low*)

Further, both rules shall carry the weight 1.0, and the OR operator employed shall be the same as it was used before.

4 The Automotive Example

Tables 1 and 2 summarize the number and amount of warranty claims issued against vehicle A shipped out in the month of January 2001. In general, most warranty claims are issued between the ages of 1 to 3 which corresponds to a mileage of 15,000 to

35,000 km. If you consider the covariate called the warranty cost, there is no linear relationship between the age of a vehicle and its mileage. In other words, older age does not mean its mileage is higher.

Table 1. Two way table of warranty counts

Division		Mileage (km)			
		0 ~ 20000	20000~40000	40000~60000	60000~80000
Age (month)	~ 12	5,179	484	1	0
	12~24	2,533	3,643	91	1
	24~36	794	2,377	1,169	308
	36~60	142	647	740	366

Table 2. Two way table of warranty costs

Division		Mileage (km)			
		0~ 20000	20000~40000	40000~60000	60000~80000
Age (month)	~ 12	238,484	21,973	81	0
	12~24	128,706	179,547	8,977	5
	24~36	26,773	70,402	32,932	5,031
	36~60	2,777	10,638	15,892	7,251

The fuzzy value corresponding to each cell in Table 3 can be obtained by multiplying the cell value by four. Table 4 represents the rule sets formed by the if-then rule.

Table 3. Fuzzy rule set

Fuzzy Rule		Mileage			
		M1	M2	M3	M4
Age	A1	C4	C1	C1	C1
	A2	C3	C4	C1	C1
	A3	C1	C2	C1	C1
	A4	C1	C1	C1	C1

According to Table 3, the number of warranty claims corresponding to the fuzzy value of the input variable, age is 5,664, 6,268, 4,648, and 1,985 and their relative ratios are 0.31, 0.34, 0.25, and 0.10. If you normalize this data with respect to A2, then you may have the weighted ratios of 0.90, 1.0, 0.74, and 0.32. Similarly, if you normalize the input variable, mileage with respect to M1, then you may have the weighted ratios of 1.0, 0.83, 0.24, and 0.08.

Fig. 2 represents the fuzzy deduction surface to which sixteen rules and normalized weighted ratios are applied. The deduction surface can be used as a type of continuous approximation distribution for Table 3. The results for deduction are as follows. If the mileage lies between 60,000 and 80,000 km, the result for fuzzy deduction is almost in accordance with the values shown in Table 3.

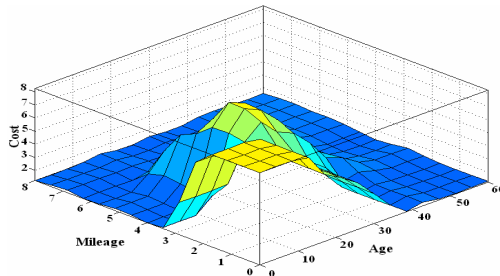


Fig. 2. Surface based on 16 rules

If the age is between 24 and 36 months while the mileage is below 60,000 km, then the cost for warranty claims is relatively low. If the age is between 12 and 24 months while the mileage is below 60,000 km, then the cost for warranty claims is relatively high.

5 Conclusion

In this analysis of warranty claims, the change in age and the limit for usage are critical elements for evaluating the effect on the number and amount of the warranty claims. According to the result of conventional studies, the accumulated ratio of the usage greatly affects the number of warranty claims. In addition, the ratio of increase in the number of claims varies with the change in age and usage while there exist various accumulated ratios of usage.

In general, when the accumulated ratio of usage is high, the expectation of the warranty claim is higher than when the ratio is low. In this paper, we proposed a fuzzy deduction method for the analysis of a two-dimensional warranty based on age and usage. In the conventional method for the estimation of the cost for warranty claims, the result is relatively higher. However, the method for the analysis of a two-dimensional warranty based on fuzzy deduction shows more reasonable results. In other words, the rate of an increase in the number of claims changes more rapidly while the accumulated ratio of usage is higher. Finally, since there exist various elements affecting warranty claims and it is necessary to perform a multi-dimensional analysis in consideration of these elements simultaneously, there is room for future improvements on this study. It is critical to probe deeply into the formation of the fuzzy groups in terms of input and output variables. In addition, it is desirable to obtain more accurate measurements for the weighted ratio of this fuzzy rule.

References

1. Kim, H.G., Rao, B.M.: Expected Warranty Cost of Two-attribute Free-replacement Warranties Based on a Bivariate Exponential Distribution. *Computers & Industrial Engineering* 38, 425–434 (2000)
2. Majeske, K.D.: A Mixture Model for Automobile Warranty Data. *Reliability Engineering and System Safety* 81, 71–77 (2003)
3. Moskowitz, H., Chun, Y.H.: A Poisson Regression Model for Two-attribute Warranty Policies. *Naval Research Logistics* 41, 355–376 (1994)
4. Pal, S., Murthy, G.S.R.: An Application of Gumbel's Bivariate Exponential Distribution in Estimation of Warranty Cost of Motor Cycles. *International Journal of Quality & Reliability Management* 20(4), 488–502 (2003)
5. Rai, B., Singh, N.: Hazard Rate Estimation from Incomplete and Unclean Warranty Data. *Reliability Engineering and System Safety* 81, 79–92 (2003)
6. Singpurwalla, N.D., Wilson, S.P.: Failure Models Indexed by Two Scales. *Advances in Applied Probability* 30, 1058–1072 (1998)

A Fuzzy Optimization Method for Multi-criteria Decision-Making Problem Based on the Inclusion Degrees of Intuitionistic Fuzzy Sets*

Changrui Yu¹ and Yan Luo²

¹ School of Information Management and Engineering,

Shanghai University of Finance and Economics, Shanghai 200433, China

²Sydney Institute of Language & Commerce, Shanghai University, 200072 Shanghai, China

{yucr, yanluo}@sjtu.edu.cn

Abstract. This paper presents a fuzzy optimization method based on the inclusion degrees of intuitionistic fuzzy sets to solve multi-criteria decision making problems under fuzzy environments. First, the inclusion degree of intuitionistic fuzzy sets is defined and a series of specific formulas of inclusion degree are presented by means of the normal implication operators. Some formulas of inclusion degree of intuitionistic fuzzy sets are generalized by defining the cardinal number of intuitionistic fuzzy sets. Then, we give multi-criteria fuzzy decision-making method based on inclusion degree of intuitionistic fuzzy sets. Finally, we illustrate the effectiveness of the method proposed in this paper by an example.

1 Introduction

Fuzzy sets were introduced by Zadeh in 1965 [1]. In the following several decades, fuzzy set theory has been used for handling fuzzy multi-criteria decision-making problems [3,4]. The main characteristic of fuzzy sets is that: the membership function assigns to each element u in a universe of discourse a membership degree ranging between 0 and 1 and the non-membership degree equals one minus the membership degree, i.e., this membership degree combines the evidence for u and the evidence against u . The single number tells us nothing about the lack of knowledge. In real applications, however, the information of an object corresponding to a fuzzy concept may be incomplete, i.e., the sum of the membership degree and the non-membership degree of an element in a universe corresponding to a fuzzy concept may be less than one. In fuzzy set theory, there is no means to incorporate the lack of knowledge with the membership degrees. A possible solution is to use intuitionistic fuzzy sets (IFSs for short), introduced by Atanassov in [5]. IFS as an extension of Zadeh fuzzy set was also applied to the decision-making problems [6-8].

In the present paper, the inclusion degree of intuitionistic fuzzy sets is applied to multi-criteria decision making in fuzzy environment. In Section 2 we define the inclusion degree of intuitionistic fuzzy sets and present a series of specific formulas of

* This research work is supported by the Natural Science Fund of China (# 70501022).

inclusion degree by means of the normal implication operators. Then Some formulas of inclusion degree of fuzzy sets are generalized to intuitionistic fuzzy sets by defining the cardinal number of intuitionistic fuzzy sets. In Section 3 we give a fuzzy method for multi-criteria decision-making method based on inclusion degree of intuitionistic fuzzy sets. Finally, we illustrate the effectiveness of the method proposed in this paper by an example.

2 Inclusion Degrees of IFSs

The concept of IFSs is an extension of Zadeh’s fuzzy sets. It give us the possibility to model unknown information by using an additional degree. In [6][14], IFSs are defined as follows:

Definition 1. An intuitionistic fuzzy set A on a universe U is defined as an object of the following form: $A = \{(u, \mu_A(u), \nu_A(u)) \mid u \in U\}$, where the functions $\mu_A : U \rightarrow [0,1]$ and $\nu_A : U \rightarrow [0,1]$ define the degree of membership and the degree of non-membership of the element $u \in U$ in A , respectively, and for every $u \in U : 0 \leq \mu_A(u) + \nu_A(u) \leq 1$.

Obviously, each ordinary fuzzy set may be written as $\{(u, \mu_A(u), 1 - \mu_A(u)) \mid u \in U\}$. For simplicity, the class of intuitionistic fuzzy sets on a universe U will be denoted by $IFS(U)$.

2.1 Inclusion Degrees Based on Implicational Operators

Definition 2. If the mapping $I : IFS(U) \times IFS(U) \rightarrow [0,1]$ satisfies

- (i) $A \subseteq B \Rightarrow I(A, B) = 1$; (ii) $I(U, \phi) = 0$;
- (iii) $A \subseteq B \subseteq C \Rightarrow I(C, A) \leq \min(I(B, A), I(C, B))$

then we call $I(A, B)$ the inclusion degree of A in B , and call I a inclusion degree function on $IFS(U)$.

Definition 3. If mapping $R : [0,1]^2 \rightarrow [0,1]$ satisfies

- (i) $R(1,0) = 0$; (ii) $R(0,0) = R(0,1) = R(1,1) = 1$

then R is called fuzzy implicational operator (briefly, implication).

Definition 4. If mapping $T : [0,1]^2 \rightarrow [0,1]$ satisfies

- (i) $T(a, b) = T(b, a)$; (ii) $a \leq c, b \leq d \Rightarrow T(a, b) = T(c, d)$;
- (iii) $T(T(a, b), c) = T(a, T(b, c))$; (iv) $T(a, 1) = a (a \in [0,1])$

then it is called triangular norm (briefly, t -norm). If T satisfies (i)-(iii) and (iv)' $T(0, a) = a (a \in [0,1])$, then we call it triangular conorm (briefly, t -conorm).

Theorem 1. Let $A, B \in IFS(U)$ and R be an implication. If R satisfies

- (i) $\forall a, b \in [0,1]$, and $a \leq b \Rightarrow R(a, b) = 1$;
- (ii) $R(a, b)$ is non-decreasing with respect to b non-increasing with respect to a ,

then the following I_1, I_2 are the inclusion degree functions of IFSs:

$$I_1(A, B) = \inf_{u \in U} [\lambda R(\mu_A(u), \mu_B(u)) + (1 - \lambda)R(v_B(u), v_A(u))], \lambda \in [0, 1], \tag{1}$$

$$I_2(A, B) = \inf_{u \in U} T(R(\mu_A(u), \mu_B(u)), R(v_B(u), v_A(u))) \tag{2}$$

where $T : [0, 1]^2 \rightarrow [0, 1]$ is a t -norm

Theorem 2. Assume that U is a finite universe and R is an implication. If R satisfies

(i) $\forall a, b \in [0, 1]$, and $a \leq b \Rightarrow R(a, b) = 1$;

(ii) $R(a, b)$ is non-decreasing with respect to b non-increasing with respect to a , then the following I_3, I_4 are the inclusion degree functions of IFSs:

$$I_3(A, B) = \frac{1}{|U|} \sum_{u \in U} [\lambda R(\mu_A(u), \mu_B(u)) + (1 - \lambda)R(v_B(u), v_A(u))], \lambda \in [0, 1] \tag{3}$$

$$I_4(A, B) = \frac{1}{|U|} \sum_{u \in U} T(R(\mu_A(u), \mu_B(u)), R(v_B(u), v_A(u))) \tag{4}$$

where $|U|$ denotes the cardinality of U and $T : [0, 1]^2 \rightarrow [0, 1]$ is a t -norm.

The implications R satisfy the conditions of above theorems, $\forall a, b \in [0, 1]$:

(i) Lukasiewicz implication: $R_L(a, b) = \min(1 - a + b, 1)$

(ii) Goguen implication: $R_\pi(a, b) = \begin{cases} 1, & \text{if } a = 0, \\ \min\left(\frac{b}{a}, 1\right), & \text{if } a > 0. \end{cases}$

(iii) Gödel implication: $R_G(a, b) = \begin{cases} 1, & \text{if } a \leq b \\ b, & \text{if } a > b. \end{cases}$

(iv) Gaines-Recher implication: $R_{GR}(a, b) = \begin{cases} 1, & \text{if } a \leq b \\ 0, & \text{if } a > b. \end{cases}$

(v) R_0 -implication: $R_\pi(a, b) = \begin{cases} 1, & \text{if } a \leq b \\ \max(1 - a, b), & \text{if } a > b \end{cases}$

2.2 Inclusion Degrees Based on the Cardinalities of Sets

Definition 5. Let U be a finite set, $A \in IFS(U)$. The cardinality of A is defined as

$$|A| = \sum_{u \in U} \frac{1 + \mu_A(u) - v_A(u)}{2} \tag{5}$$

It is easy to prove that the following inclusion functions of fuzzy sets $I_5 - I_{10}$ [9] still hold to IFSs.

Theorem 3. Let U be a finite set, $A \in IFS(U)$. Then the following $I_5 - I_{10}$ are inclusion degree functions of IFSs:

$$\begin{aligned}
 I_5(A, B) &= \begin{cases} 1, & A = \phi \\ \frac{|A \cap B|}{|A|}, & A \neq \phi \end{cases}; & I_6(A, B) &= \begin{cases} 1, & A = B = \phi \\ \frac{|B|}{|A \cup B|}, & \text{otherwise} \end{cases}; \\
 I_7(A, B) &= \begin{cases} 1, & B = U \\ \frac{|A^c \cap B^c|}{|B^c|}, & B \neq U \end{cases}; & I_8(A, B) &= \begin{cases} 1, & A = B = U \\ \frac{|A^c|}{|A^c \cup B^c|}, & \text{otherwise} \end{cases}; \\
 I_9(A, B) &= \frac{|A^c \cup B|}{|A^c \cup A \cup B \cup B^c|}; & I_{10}(A, B) &= \begin{cases} 1, & A = \phi \text{ or } B = U \\ \frac{|A^c \cap A \cap B \cap B^c|}{|A \cup B^c|}, & \text{otherwise} \end{cases}.
 \end{aligned}$$

2.3 Generation of Inclusion Degrees

Theorem 4. Let I be an inclusion degree function on $IFS(U)$ and mapping $h : [0,1]^2 \rightarrow [0,1]$ satisfies

(i) $h(0,0) = 0, h(1,1) = 1$; (ii) $h(a, b)$ is non-decreasing with respect to a and b , then $GI(A, B) = h(I(A, B), I(B^c, A^c)) (\forall A, B \in IFS(U))$ is inclusion degree of A in B , and GI is an inclusion degree on $IFS(U)$.

Theorem 5. Let I_1, I_2 be inclusion degree functions on $IFS(U)$, and $h : [0,1]^2 \rightarrow [0,1]$ satisfies

(i) $h(0,0) = 0, h(1,1) = 1$; (ii) $h(a, b)$ is non-decreasing with respect to a and b , then $GI(A, B) = h(I_1(A, B), I_2(A, B)) (\forall A, B \in IFS(U))$ is inclusion degree of A in B , and GI is an inclusion degree on $IFS(U)$.

3 Multi-criteria Fuzzy Decision-Making Based on Inclusion Degrees

Definition 6 (Multi-criteria fuzzy decision-making problem [7]). Let M be a set of alternatives and let C be a set of criteria, where $M = \{M_1, M_2, \dots, M_m\}$, $C = \{C_1, C_2, \dots, C_n\}$.

Assume that the characteristics of the alternative M_i are presented by the IFS shown as follows: $M_i = \{(C_1, \mu_{i1}, \nu_{i1}), (C_2, \mu_{i2}, \nu_{i2}), \dots, (C_n, \mu_{in}, \nu_{in})\}$, where μ_{ij} indicates the degree to which the alternative M_i satisfies criterion C_j , ν_{ij} indicates the degree to which the alternative M_i does not satisfy criterion $C_j (\mu_{ij}, \nu_{ij}) \in L^* (j = 1, 2, \dots, n; i = 1, 2, \dots, m)$.

Assume that there is a decision-maker who wants to choose an alternative which satisfies the criteria C_j, C_k, \dots , and C_p or which satisfies the criterion C_s . This decision-maker's requirement is represented by the following expression: C_j and C_k and \dots and C_p or C_s .

It is noted that we say an alternative satisfies a criterion if it meets some desirable level of an evaluation criterion. The satisfaction is gradual and is characterized by a dual information: a degree of satisfaction and a degree of non-satisfaction.

We now use the inclusion degrees of IFSs to solve the multi-criteria fuzzy decision-making problem. The basic idea is similar to the TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) [10,11]. Firstly, the ideal solution and the anti-ideal solution are constructed, where the ideal solution and the anti-ideal solution are respectively the best and the worst solution supposed but not existing in the set of alternatives. Then we compare the inclusion degrees of the ideal solution in alternative M_i and the inclusion degrees of alternative M_i in the anti-ideal solution. The alternative containing the ideal solution maximally as well as being contained by the negatively ideal solution minimally is the best choice. For this purpose, we introduce the following definition.

Definition 7. Let $M_i = (i = 1, \dots, m)$ be the set of alternatives and $C_j = (j = 1, \dots, n)$ be the set of criteria,

(i) The ideal solution and the anti-ideal solution satisfying the criteria C_j, C_k, \dots, C_p are defined as follows:

$$G_1 = \left\{ \left(C_j, \bigvee_{i=1}^m \mu_{ij}, \bigwedge_{i=1}^m v_{ij} \right), \left(C_k, \bigvee_{i=1}^m \mu_{ik}, \bigwedge_{i=1}^m v_{ik} \right), \dots, \left(C_p, \bigvee_{i=1}^m \mu_{ip}, \bigwedge_{i=1}^m v_{ip} \right) \right\} \tag{6}$$

$$= \left\{ (C_j, \mu_{gj}, v_{gj}), (C_k, \mu_{gk}, v_{gk}), \dots, (C_p, \mu_{gp}, v_{gp}) \right\}$$

$$B_1 = \left\{ \left(C_j, \bigwedge_{i=1}^m \mu_{ij}, \bigvee_{i=1}^m v_{ij} \right), \left(C_k, \bigwedge_{i=1}^m \mu_{ik}, \bigvee_{i=1}^m v_{ik} \right), \dots, \left(C_p, \bigwedge_{i=1}^m \mu_{ip}, \bigvee_{i=1}^m v_{ip} \right) \right\} \tag{7}$$

$$= \left\{ (C_j, \mu_{bj}, v_{bj}), (C_k, \mu_{bk}, v_{bk}), \dots, (C_p, \mu_{bp}, v_{bp}) \right\}$$

(ii) The ideal solution and the negatively ideal solution satisfying the criteria C_s are defined as follows:

$$G_2 = \left\{ \left(C_s, \bigvee_{i=1}^m \mu_{is}, \bigwedge_{i=1}^m v_{is} \right) \right\} = \left\{ (C_s, \mu_{gs}, v_{gs}) \right\} \tag{8}$$

$$B_2 = \left\{ \left(C_s, \bigwedge_{i=1}^m \mu_{is}, \bigvee_{i=1}^m v_{is} \right) \right\} = \left\{ (C_s, \mu_{bs}, v_{bs}) \right\} \tag{9}$$

Definition 8. The inclusion degree $D(M_i)$ of the ideal solution in alternative M_i and the inclusion degree $d(M_i)$ of alternative M_i in the anti-ideal solution are respectively defined as follows:

$$D(M_i) = \max(I(G_1, M_{i1}), I(G_2, M_{i2})) \tag{10}$$

$$d(M_i) = \min(I(M_{i1}, B_1), I(M_{i2}, B_2)) \tag{11}$$

where I denotes the inclusion degree function, $M_{i1} = \{(C_j, \mu_{ij}, \nu_{ij}), (C_k, \mu_{ik}, \nu_{ik}), \dots, (C_p, \mu_{ip}, \nu_{ip})\}$, $M_{i2} = \{(C_s, \mu_{is}, \nu_{is})\}$, $i = 1, \dots, m$.

Definition 9. The ranking index of alternative $M_i = (i = 1, \dots, m)$ is defined as follows:

$$P_i = \frac{D(M_i)}{d(M_i) + D(M_i)} \tag{12}$$

The procedure of solving multi-criteria fuzzy decision-making problem (definition 0.2.4) is as follows:

(i) calculate the ideal solution G_1 and the anti-ideal solution B_1 satisfying the criteria C_j, C_k, \dots, C_p , calculate the ideal solution G_2 and the anti-ideal solution B_2 satisfying the criteria C_s ;

(ii) calculate the inclusion degree $I(G_j, M_{ij})$ of G_j in M_{ij} and the inclusion degree $I(M_{ij}, B_j)$ of M_{ij} in B_j , $j = 1, 2; i = 1, \dots, m$;

(iii) calculate the inclusion degree $D(M_i)$ of the ideal solution in alternative M_i and the inclusion degree $d(M_i)$ of M_i in the anti-ideal solution;

(iv) calculate the ranking index p_i of alternative $M_i = (i = 1, \dots, m)$

(v) if there exists $i_0 \in \{1, 2, \dots, m\}$ such that $p_{i_0} = \max(p_1, p_2, \dots, p_m)$, then alternative M_{i_0} is the best choice.

The reason for introducing the ideal solution and the anti-ideal solution simultaneously in above method is that when two alternatives contain the ideal solution by the same inclusion degree, we introduce the negatively ideal solution for differentiating which alternative is superior. Then the alternative contained by the anti-ideal solution with the less inclusion degree is the better choice.

4 An Example

Let M_1, M_2, M_3, M_4, M_5 be five alternative, and let C_1, C_2, C_3 be three criteria. Assume that the characteristics of the alternatives are represented by the IFSSs shown as follows:

$$\begin{aligned}
 M_1 &= \{(C_1, (0.2,0.2)), (C_2, (0.3,0.1)), (C_3, (0.2,0))\} \\
 M_2 &= \{(C_1, (0.3,0.3)), (C_2, (0.2,0.2)), (C_3, (0.3,0.1))\} \\
 M_3 &= \{(C_1, (0.4,0.4)), (C_2, (0.5,0.4)), (C_3, (0.3,0.2))\} \\
 M_4 &= \{(C_1, (0.5,0.3)), (C_2, (0.4,0.4)), (C_3, (0.5,0.3))\} \\
 M_5 &= \{(C_1, (0.4,0.4)), (C_2, (0.6,0.3)), (C_3, (0.6,0.4))\}
 \end{aligned}$$

and assume that the decision-maker wants to choose an alternative which satisfies the criteria C_1, C_2 or which satisfies the criterion C_3 .

Firstly, we construct the ideal solution and the negatively ideal solution satisfying the criteria C_1, C_2 and satisfying criterion C_3 .

$$\begin{aligned}
 G_1 &= \{(C_1, 0.5, 0.2), (C_2, 0.6, 0.1)\}, G_2 = \{(C_3, 0.6, 0)\} \\
 B_1 &= \{(C_1, 0.2, 0.4), (C_2, 0.2, 0.4)\}, B_2 = \{(C_3, 0.2, 0.4)\}
 \end{aligned}$$

Take I_3 in theorem 2 as the inclusion degree function I and take $\lambda = \frac{1}{2}$, i.e.,

$$I(A, B) = \frac{1}{|U|} \sum_{u \in U} \frac{R(\mu_A(u_i), \mu_B(u_i)) + R(v_B(u_i), v_A(u_i))}{2}$$

where we choose Lukasiewicz implication R_L as implication R , i.e., $R_L(a, b) = \min(1 - a + b, 1)$, $\forall a, b \in [0, 1]$. So we obtain the inclusion degree $I(G_j, M_{ij})$ of G_j in M_{ij} and the inclusion degree $I(M_{ij}, B_j)$ of M_{ij} in B_j ($j = 1, 2; i = 1, 2, 3, 4, 5$) listed in table 1 and 2.

Table 1. Inclusion degrees of G_1 in M_{i1} and M_{i1} in B_1

	M_{11}	M_{21}	M_{31}	M_{41}	M_{51}
$I(G_1, M_{i1})$	0.85	0.8	0.825	0.85	0.875
$I(M_{i1}, B_1)$	0.8	0.9	0.875	0.85	0.825

Table 2. Inclusion degrees of G_2 in M_{i2} and M_{i2} in B_2

	M_{12}	M_{22}	M_{32}	M_{42}	M_{52}
$I(G_2, M_{i2})$	0.8	0.8	0.75	0.8	0.8
$I(M_{i2}, B_2)$	0.8	0.8	0.85	0.8	0.8

Using formulae (5) and (6) we get the inclusion degree $D(M_i)$ of the ideal solution in alternative M_i and the inclusion degree $d(M_i)$ of M_i in the negatively ideal solution. We list them in table 3.

From (7), we get the ranking index of alternatives as follows:

$$p_1 = 0.515, p_2 = 0.5, p_3 = 0.4925, p_4 = 0.515, p_5 = 0.5224$$

Therefore, alternative M_5 is the best choice.

Table 3. Inclusion degrees of G_2 in M_{i2} and M_{i2} in B_2

	M_1	M_2	M_3	M_4	M_5
$D(M_i)$	0.85	0.8	0.825	0.85	0.875
$d(M_i)$	0.8	0.8	0.85	0.8	0.8

5 Conclusions

This paper presents a fuzzy optimization method based on the inclusion degrees of intuitionistic fuzzy sets to solve multi-criteria decision making problems under fuzzy environments. Inclusion degree is a quantity describing that a set is contained by another set and is quantitative description of containment relation. It holds the uncertainty of the relation. The inclusion degree theory and IFS theory are the important tools in studying the uncertain knowledge. The proposed method in the paper has been demonstrated by an example, illustrating the power of the approach to solve multi-criteria fuzzy decision making problems. This research work not only develops and enriches the fundamental theory of IFSs but also provides a new idea for the applications of IFS theory.

References

1. Zadeh, L.A.: Fuzzy Sets. *Information and Control* 8(3), 338–356 (1965)
2. Tran, L., Duckstein, L.: Comparison of Fuzzy Numbers Using a Fuzzy Distance Measure. *Fuzzy Sets and Systems* 130(3), 331–341 (2002)
3. Wang, X.Z., Kerre, E.E.: Reasonable Properties for the Ordering of Fuzzy Quantities (I) and (II). *Fuzzy Sets and Systems* 118(3), 375–385 (2001)
4. Atanassov, K.: Intuitionistic Fuzzy Sets. VII ITKR's Session, Sofia (1983)
5. Atanassov, K., Pasi, G., Yager, R.: Intuitionistic Fuzzy Interpretations of Multi-Measurement Tool Multi-Criteria Decision Making. In: Kacprzyk, J., Atanassov, K. (eds.) *Proceedings of the Sixth International Conference on Intuitionistic Fuzzy Sets. Notes on Intuitionistic Fuzzy Sets*, vol. 8(3), pp. 66–74 (2002)
6. Atanassov, K., Pasi, G., Yager, R., Atanassova, V.: Intuitionistic Fuzzy Group Interpretations of Multi-Person Multi-Criteria Decision Making. In: *Proceedings of the Third Conference of the European Society for Fuzzy Logic and Technology EUSFLAT 2003, Zittau*, pp. 177–182 (2003)
7. Pasi, G., Yager, Y., Atanassov, K.: Intuitionistic Fuzzy Graph Interpretations of Multi-Person Multi-Criteria Decision Making: Generalized Net Approach. In: *Proceedings of 2004 second International IEEE Conference Intelligent Systems*, vol. 2, pp. 434–439 (2004)
8. Atanassov, K.: *Intuitionistic Fuzzy Sets*. Physica-Verlag, Heidelberg (1999)
9. Fan, J., Xie, W., Pei, J.: Subsethood Measure: New Definitions. *Fuzzy Sets and Systems* 106, 201–209 (1999)
10. Chen, C.T.: Extensions of the TOPSIS for Group Decision-Making Under Fuzzy Environment. *Fuzzy Sets and Systems* 114(1), 1–9 (2000)
11. Chu, T.C., Lin, Y.C.: A Fuzzy TOPSIS Method for Robot Selection. *The International Journal of Advanced Manufacturing Technology* 21, 284–290 (2003)

A GP Based Approach to the Classification of Multiclass Microarray Datasets

Chun-Gui Xu^{1,2} and Kun-Hong Liu¹

¹Hefei Institute of Intelligent Machines, Chinese Academy of Sciences, P.O. Box 1130, Hefei, Anhui, 230031, China

²Department of Biology, University of Science and Technology of China, Hefei, Anhui, 230026, China

{xcg1984, lkkqz}@mail.ustc.edu.cn

Abstract. In this paper, we propose a genetic programming (GP) based approach to analyze multiclass microarray datasets. Here, a multiclass problem is divided into a set of two-class problems. Instead of applying a tree for each two-class problem, a small-scale ensemble system containing a set of trees is deployed and denoted by sub-ensemble (SE). The SEs tackling the respective two-class problems are combined to construct an individual of the GP, so that an individual can deal with a multiclass problem directly. In the experiments, the GP implements classification and feature selection at the same time. The results obtained at independent test sets show that our method is efficient in the search of genes with great biological significance, and achieves high classification accuracy at the same time.

Keywords: Genetic Programming; Sub-ensemble.

1 Introduction

With the development of microarray technology, it is possible to diagnose and classify some particular cancers directly based on DNA microarray datasets.

In this paper, we focus the discussion on the analysis of multiclass microarray datasets. Although there are many evolutionary based approaches proposed to analyze the multiclass gene expression data successfully, most of these methods are implemented by using an evolutionary algorithm (EA) to select an optimal gene subset firstly, and then solving the classification task with a multiclass classifier[1-3]. However, when there are a large number of classes with a very small sample size in each class, it is very hard to obtain a truly optimal gene subset for distinguishing all the classes. On the contrary, it is much easier to obtain a set of genes aiming to discriminate one class from all other classes. So we use the class-decomposition based solution for the classification of multiclass microarray datasets.

GP has been widely applied to deal with classification problems because it can discover underlying data relationships and present them mathematically. Among the evolutionary approaches, GP is a promising solution for the discovery of potentially important gene expression profiling data because it can generate comprehensible rules for classification from microarray datasets owing to its powerful search ability [4-5].

Although these methods had been proved to be successful, they are only applicable to two-class microarray datasets. It should be noted that the fusion of multiple two-class classifiers is different from solving a set of two-class problems, and some efficient methods are required for the combination of the classifiers aiming at respective two-class problems. For example, some classifiers may claim that a sample belongs to their own class at the same time, which results in a confliction case.

In this paper, we try to implement an efficient GP based approach for the analysis of multiclass microarray datasets. The advantage of this GP is that it can solve the feature selection and classification simultaneously, and also provide deep insight of the biological significance of genes in each sub-class.

2 Method

GP was introduced by Koza [6]. It is a branch of genetic algorithm (GA). GP's individuals are trees, in which the leaf nodes and non-leaf nodes are chosen from the terminal and function sets respectively. Usually, in the feature selection problem, the terminal set consists of features and constants, and the function set consists of several mathematical or logical functions, which connect different terminals.

A typical GP's tree can't be used to tackle with multiclass problem directly. So some solutions were proposed to extend the application of traditional GP to multiclass classification problem [7, 8]. In these papers, the multiclass problem is formulated as multiple two-class problems, each of which is handled by a tree. In this way, the i -th two-class problem can also be regarded as the i -th class classification task, and then a tree group is used to distinguish the two respective sample groups. So for the datasets with n -class, a tree group containing n trees is needed to classify one class from other classes. This schemes works well for many classification tasks, however, due to the small sample size problem, it is not easy to obtain a predictive tree with robust performance for a two-class problem in multiclass microarray data classification task. Consequently, the combination of tree group may produce worse results unavoidably. What's more, the classification task is accompanied by the unbalanced data problem, which makes the classification task of the first group more difficult.

Here, we propose an efficient GP based scheme for multiclass microarray dataset classification along with an efficient solution for the confliction cases.

An ensemble system has been proved to be more accurate and robust than an excellent single classifier in many fields [9]. Based on this theory, instead of applying a single tree to a two-class problem, an ensemble of k trees is used. In this way, it is hopeful that even when a tree can't distinguish a 'hard' sample, other trees in the ensemble will recognize it, so that the final ensemble can produce a correct output. These ensemble systems are named as sub-ensemble (SE) systems for the clarification of their roles. Consequently, for an n -class microarray dataset, n SEs solving the respective two-class problems are required to build a multiclass classifier.

In this GP, a population is used to tackle a two-class problem. Here, each individual is a SE, so n runs are required to obtain a multiclass classifier. After n runs, the individuals with highest fitness values are combined to construct the final multiclass classifier.

10-fold cross validation (10-fold CV) is applied to evaluate the generalization ability of the trees in each SE system. Then the CV accuracies are assigned as weights of the respective trees, and the output of a SE is based on the weighted majority vote of k trees in it. Assuming that in the i -th run, the m -th SE is denoted by SE_m^i . Assuming that tree a in SE_m^i is denoted by $T_{i,m}^a$, and its 10-fold CV accuracy is $AC_{i,m}^a$, then its weight $w_{i,m}^a$ is calculated by $w_{i,m}^a = AC_{i,m}^a / \sum_{p=1}^k AC_{i,m}^p$. So the final output of the SE_m^i is $O_{i,m}$, where $O_{i,m} = \sum_{p=1}^k w_{i,m}^p O_{Ti,m}^p$, and $O_{Ti,m}^p$ is the output of the corresponding tree. The sample is assigned to the i -th class only when $O_{i,m}$ is larger than 0.

A weight-based scheme is used to overcome the skewness among data here. Assuming that there are equal samples in each class in a dataset, and after dividing this dataset into n two-class problems, there are S_{i1} samples in G_{i1} , and S_{i2} samples in G_{i2} for the i -th two-class problem. The weights for the correct classification of the samples in two groups are:

$$\begin{cases} W_{i1} = S_{i2} / (S_{i1} + S_{i2}) \\ W_{i2} = S_{i1} / (S_{i1} + S_{i2}) \end{cases} \quad (1)$$

If SE_m^i correctly classifies l_{i1} samples in G_{i1} , and l_{i2} samples in G_{i2} , its covering score is:

$$C_m^i = (W_{i1} \times l_{i1} + W_{i2} \times l_{i2}) / (W_{i1} \times S_{i1} + W_{i2} \times S_{i2}) \quad (2)$$

Then if SE_m^i correctly classifies a sample in G_{i1} , it gets a higher score than in G_{i2} . As a result, SE_m^i is encouraged to distinguish more samples in G_{i1} . This weighed scheme is simple and can efficiently guide the SEs to evolve towards a balanced covering of classes.

This covering score is also regarded as the weight of the corresponding SE. When combining the SEs to a multiclass classifier, the confliction case would occur, that is, some SEs in a classifier decide a sample belonging to their classes at the same time. In this case, the covering scores of these SEs are compared, and the output of a SE with the highest weight in the conflicting group will be selected as the final decision.

The rejection case occurs only when all SEs can't recognize a sample as its own class. In such case, this sample is also rejected here. It should be noted that as each SE contains different trees, the probability of the rejection case occurring is very small even when the size of SE is set to 3. So no special solutions are required for this problem.

The Ramped Half-and-Half method proposed in [6] is applied here to generate the first generation, and the dynamic maximum tree depth technique proposed in [10] is used to control bloat here. The standard crossover and mutation operator are deployed here. In the crossover process, two trees are randomly selected from individuals. Then two random nodes are chosen from both parent trees, and the respective branches are swapped so as to create two offspring. The mutation operator randomly chooses a tree from individuals, and selects a node from the parent tree at random and replaces it with a new randomly generated sub-tree. As the task of the GP is to discover

potentially important genes in a huge number of candidates, the mutation operator is important to enhance the exploration in the huge search space

The generalization ability of an individual is directly connected with the performance of all SEs in it, and it is robust when all SEs get high covering scores. So for each individual, its fitness value is calculated by

$$F_m = \sum_{i=1}^n C_m^i / n \quad (3)$$

With this design, the SE with higher fitness is preferred. Due to the huge search space, only the SEs with top fitness values are kept so as to accelerate the evaluation process. When two or more SEs are assigned to the same fitness score, the one containing fewer features will be selected firstly.

As there are many feature subsets generated in parallel during evolution, a set of globally optimal or at least near optimal trees can be obtained for each two-class problem.

3 Results

Three microarray datasets are deployed here: NCI60 [11] and Leukemia [12]. The pre-processing is exactly the same as described in [2], and only the truncated data with 61 samples and 1000 genes is used. For Leukemia, the raw data is transformed to natural logarithmic values, and then standardizing each sample to zero mean and unit variance. After normalization, all of the 2308 genes are kept for evaluation.

In all experiments, we run GP 10 times with random initialization. In each run, the corresponding parameters of the GP are listed in Table 1.

Table 1. The settings of primary GP parameters

Parameter	Setting
Terminal set (T)	All gene expression values and constant values
Function set (F)	Boolean and floating point operators: gt(>),le(<=),times (*),minus(-),plus(+), max, min.
Crossover probability	0.8
Mutation probability	0.3
Termination criteria	Fitness score reaches 1 or running 200 generations
Dynamic maximum tree depth limit	5
Strict depth limit	10

NCI60 is hard to deal with. Due to the small sample size in each class, some authors did not split the dataset into training and test datasets, and only test the methods on the whole dataset using Leave-One-Out Cross-Validation (LOOCV) to evaluate the performance of their methods [1, 2]. On the contrary, our method aims at obtaining the classification results on the independent test datasets with high accuracy. The results obtained by other researchers are listed in Table 2, and our method takes obvious advantages. For the results obtained by [2, 13], although the authors also used a truncated dataset with 61 samples and 1000 genes, they used 41 samples for training and 20 for testing. As the size of our training and test sets are 40 and 21, respectively, our results can not be compared with theirs directly. But since our LOOCV and independent test accuracies are higher than theirs with smaller

training set, we believe that our method is more robust. At the same time, it should be pointed out that the results obtained in [3] may be too optimistic because their error rate estimator is different from the widely deployed ones, and the test set was involved in the fitness evaluation. So we believe that our method can achieve the best performance in this case. When setting the SE scale to 1, that is, only using a tree to deal with a two-class problem, the best results obtained is 80.95%, not better than those of other studies. However, only by setting the size of SE to 3, the best individual obtained is much better than other studies.

Table 2. Comparisons of the accuracies on the NCI60 datasets. Here, A_{cv} is the LOOCV accuracy, and A_t is the accuracy on independent test set. N/A means not available.

Research	A_{cv}	A_t
[1]	76.23	N/A
[2]	88.52	N/A
[3]	85.37	95
[13]	87.8	85
This study	92.31	95.24

When setting the SE scale to 1, the best accuracy on test set can only reach 97.06% on the Leukemia dataset. It is because for the first and third two-class problem, none of the obtained trees can classify all samples. But even when setting the SE size to 3, we can again obtain 100% accuracy without requiring more accurate trees. It is because the trees in a same SE do not wrongly distinguish a sample simultaneously, which proves the efficiency of this individual design.

Z-score is a widely deployed method to evaluate the frequency of different genes. Let $Z = [X_i - E(X_i)] / \sigma$, where X_i is the frequency that gene i is selected, $E(X_i)$ is the expected number of times gene i is selected, and σ is the standard deviation of variance. Let p_i denote the probability of gene i being selected randomly, which is

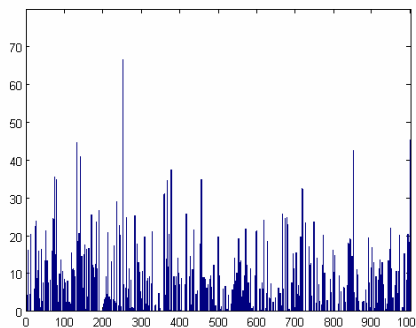


Fig. 1. The genes' z-score value on NCI60 datasets. X-axis represents the gene index, and y-axis represents the corresponding z-scores.

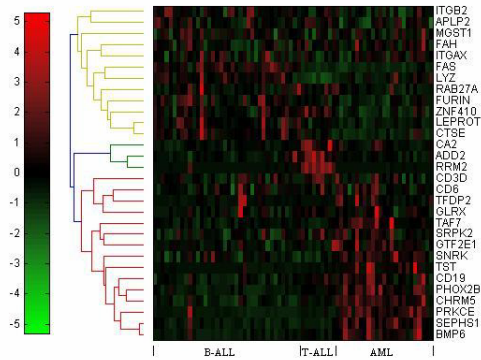


Fig. 2. The heat map of the expressions of top30 genes in leukemia dataset. On the left is the color bar. Red means that this gene is up-regulated, and green otherwise. The tree-like structure on the left of the heat map is hierarchical clustering of these genes whose names are listed on the right. The 3 classes of leukemia are: B-ALL, T-ALL and AML.

approximately equal to the total counts of frequency in a population divided by the number of individuals in the population, then divided by the total number of genes. Let A denote the number of individuals, then $E(X_i) = A \times p_i$, and $\sigma = \sqrt{A \times p_i \times (1 - p_i)}$.

Fig. 1 shows the z-score of the total frequency of each gene occurring in the final generation of ten runs obtained by setting the size of SE to 3 on NCI60 dataset. All genes have been selected more than once in this case. The reason lies in that there are nine SEs in each individual for the nine categories. As a result, there are 27 thousand trees in the experiments. As each tree contains at least two features, all features stand a chance to be selected. But it is apparent that there are always some dominant genes in the gene selection process. In contrast, only 5270 out of 7129 genes (73.9%) appear in all experiments. And in different datasets, the range of z-score value will be quite different.

4 Conclusions

We have proposed a GP based approach to deal with the gene selection and classification task based on multiclass microarray datasets. The advantage of the multiclass classifier built by GP the small-scale ensemble systems, which guarantees robust results. With this method, important genes can be selected as important references for clinic diagnosis or cancer development. At the same time, high classification accuracies have been achieved by comprehensive rules constructed by genes. The experimental results prove that our approach is powerful even when analyzing the datasets with very small sample sizes. In conclusion, we hope our ensemble method could be helpful for discovery of tumor-related genes and auxiliary approach to diagnose tumors.

References

1. Jirapech-Umpai, T., Aitken, S.: Feature Selection and Classification for Microarray Data Analysis: Evolutionary Methods for Identifying Predictive Genes. *BMC Bioinformatics* 6, 148 (2005)
2. Liu, J.J., Cutler, G., Li, W.X., Pan, Z., Peng, S.H., Hoey, T., Chen, L.B., Ling, X.F.B.: Multiclass Cancer Classification and Biomarker Discovery Using GA-based Algorithms. *Bioinformatics* 21, 2691–2697 (2005)
3. Ooi, C.H., Tan, P.: Genetic Algorithms Applied to Multi-class Prediction for the Analysis of Gene Expression Data. *Bioinformatics* 19, 37–44 (2003)
4. Hong, J.H., Cho, S.B.: The Classification of Cancer Based on DNA Microarray Data that Uses Diverse Ensemble Genetic Programming. *Artificial Intelligence in Medicine* 36, 43–58 (2006)
5. Yu, J.J., Yu, J.D., Almal, A.A., Dhanasekaran, S.M., Ghosh, D., Worzel, W.P., Chinnaiyan, A.M.: Feature Selection and Molecular Classification of Cancer Using Genetic Programming. *Neoplasia*, 9, 292–U216 (2007)
6. Koza, J.R.: *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge (1992)
7. Kishore, J.K., Patnaik, L.M., Mani, V., Agrawal, V.K.: Application of Genetic Programming for Multicategory Pattern Classification. *IEEE Transactions on Evolutionary Computation* 4, 242–258 (2000)
8. Muni, D.P., Pal, N.R., Das, J.: A Novel Approach to Design Classifiers Using Genetic Programming. *IEEE Transactions on Evolutionary Computation* 8, 183–196 (2004)
9. Kuncheva, L.I.: *Combining Pattern Classifiers: Methods and Algorithms*. Wiley, Chichester (2004)
10. Silva, S., Almeida, J.: Dynamic Maximum Tree Depth – a Simple Technique for Avoiding Bloat in Tree-based GP. In: *GECCO 2003*, pp. 1776–1787 (2003)
11. Ross, D.T., Scherf, U., Eisen, M.B., Perou, C.M., Rees, C., Spellman, P., Iyer, V., Jeffrey, S.S., Van de Rijn, M., Waltham, M., Pergamenschikov, A., Lee, J.C.E., Lashkari, D., Shalon, D., Myers, T.G., Weinstein, J.N., Botstein, D., Brown, P.O.: Systematic Variation in Gene Expression Patterns in Human Cancer Cell Lines. *Nat. Genet.* 24, 227–235 (2000)
12. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S.: Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. *Science* 286, 531–537 (1999)
13. Lin, T.C., Liu, R.S., Chen, C.Y., Chao, Y.T., Chen, S.Y.: Pattern Classification in DNA Microarray Data of Multiple Tumor Types. *Pattern Recognition* 39, 2426–2438 (2006)

A Novel Approach for Blind Channel Equalization

A. Özen¹, A. Güner¹, O. Çakır¹, E. Tuğcu¹, B. Soysal², and I. Kaya¹

¹ Department of Electrical and Electronics Engineering
Karadeniz Technical University TI DSP LAB., Trabzon / Turkey

² Department of Electrical and Electronics Engineering
Atatürk University, Erzurum / Turkey
{aliozen, guner, cakir, emintugcu, ikaya}@ktu.edu.tr,
bsoysal@atauni.edu.tr

Abstract. The Constant Modulus Algorithm (CMA), while the most commonly used blind equalization technique, converges very slowly. The convergence rate of the CMA is quite sensitive to the adjustment of the step size parameter used in the update equation as in the Least Mean Squares (LMS) algorithm. A novel approach in adjusting the step size of the CMA using the fuzzy logic based outer loop controller is presented in this paper. It also presents a computational study and simulation results of this newly proposed algorithm compared to other variable step size CMA such as conventional CMA, Normalized CMA (N-CMA) [1], Modified CMA (M-CMA) [2], CMA-Soft Decision Directed (CMA-SDD) [3]. The simulation results have demonstrated that the proposed algorithm has considerably better performance than others.

Keywords: Blind channel equalization, fuzzy step size, constant modulus algorithm, adaptive training.

1 Introduction

Adaptive equalizers are usually employed in communication systems to compensate the Intersymbol Interference (ISI) due to a frequency selective channel response. Conventional adaptive equalizers periodically use the training sequences to prevent catastrophic error propagation. Therefore, a portion of transmission bandwidth is wasted. In order to use transmission bandwidth more efficiently, blind equalization techniques allow the estimation of the channel or the equalizer based only on the received signal with no training symbols. While many algorithms have been proposed in the last twenty years, due to its simplicity, good performance, and robustness, the CMA [4, 5] is widely used in practice. Perhaps the greatest drawback of the CMA is its relatively slow convergence, which becomes ever more significant as wireless applications involving rapid changes in channel characteristics become more prominent [6].

Blind adaptive filtering using the CMA [4, 5] utilizes a constant step size to change its coefficients in response to the changing environment. Using a large step size will cause a fast initial convergence, but result in larger fluctuation in the steady state; the results are opposite when a small step size is used. Therefore, the choice of the step

size reflects a trade-off between misadjustment and speed of convergence. The convergence rate of the CMA is quite sensitive to the step size parameter used in the update equation. There are many methods available to select different step size for the CMA during the different stages of adaptation. Among them the most commonly used is the automatic switching scheme [7] that utilizes a large step size during the transient state and switches to a smaller step size during the steady state. Other step size variation methods include:

(i) Douglas L. Jones [1] controls the step size, for efficient implementation, by using the signal vector energy, $\|x_n\|^2$, is computed recursively as in the normalized LMS algorithm.

(ii) Chahed et al. [2] adjust the step size by using a time varying step size parameter depending upon the squared Euclidian norm of the channel output vector and on the equalizer output.

(iii) De Castro et al. [8] proposed a concurrent CMA and Decision Directed (DD) blind equalization scheme. Specifically, let $w = w_c + w_d$, where w_c is the weight vector of the CMA equalizer which is designed to minimize the CMA cost function and w_d is the weight vector of the DD equalizer which is designed to minimize the decision based mean square error (MSE).

(iv) Chen et al. [3] proposed an alternative scheme that operates a CMA equalizer and a Soft Decision Directed (SDD) equalizer concurrently. The CMA part is identical to that of the concurrent CMA and DD scheme. The SDD equalizer is designed to maximize log of the local a posteriori probability density function (PDF) criterion by adjusting w_d using a stochastic gradient algorithm.

However, most of these approaches involve significant increases in complexity or computational cost. The aim of this paper is to present a low complexity and high performance blind equalization by the proposed fuzzy outer loop controller. So for improving the convergence speed, a Fuzzy Logic based outer loop controller which works independently from the nature of the system is used in order to change the step size of the CMA. The proposed algorithm is compared with related blind algorithms, and simulation results confirm that the proposed algorithm leads to enhanced performance.

2 Blind Channel Equalization

The baseband model of a digital communication channel can be characterized by a symbol-spaced Finite Impulse Response (FIR) filter and additive white Gaussian noise (AWGN) source. The received signal at sample k is given by

$$v(k) = \sum_{i=0}^{n-1} h_i x(k-i) + \eta(k) \quad (1)$$

where, n is the length of the channel impulse response (CIR), $h_i = h_{iR} + jh_{iI}$ are the complex channel taps, the complex symbol sequence $x(k) = x_R(k) + jx_I(k)$ is assumed to be independent identically distributed (iid), $\eta(k) = \eta_R(k) + j\eta_I(k)$ is an iid AWGN with $E[\eta_R^2(k)] = E[\eta_I^2(k)] = \sigma_\eta^2$, and $E[\cdot]$ denotes the expectation.

A symbol-space equalizer is employed, which has an FIR structure defined by

$$y(k) = \sum_{i=0}^{m-1} w_i v(k-i) = W^T V \tag{2}$$

where, m is the equalizer order, $W(k) = [w_0 \ w_1 \ \dots \ w_{m-1}]^T$ is the equalizer weight vector with $w_i = w_{iR} + jw_{iI}$, and $v(k) = [v(k) \ v(k-1) \ \dots \ v(k-m+1)]^T$ the equalizer input vector. According to [3] for dealing with non-minimum phase delay channel profiles, a central tap should be defined by having a unity gain at the tap $k_d \approx m/2$ before starting blind training. Therefore, the equalizer weights are initialized to $w_i = 1 + j0$ for $i = k_d$ and $w_i = 0 + j0$ for $i \neq k_d$.

2.1 The Constant Modulus Algorithm

The CMA adjusts the equalizer weights by minimizing the cost function

$$\bar{J}_{CMA}(W) = E[(|y(k)|^2 - \Delta_2)^2] \tag{3}$$

using a stochastic gradient algorithm, where Δ_2 is a real positive constant defined by

$$\Delta_2 = E[|x(k)|^4] / E[|x(k)|^2]^2. \tag{4}$$

At sample k , given $y(k) = W^T V$, the CMA adapts W according to [4, 5]

$$\left. \begin{aligned} e(k) &= y(k)(\Delta_2 - |y(k)|^2) \\ w(k+1) &= w(k) + \mu e(k)v^*(k) \end{aligned} \right\} \tag{5}$$

where, μ is a step size parameter and $v^*(k)$ is the complex conjugate of $v(k)$.

Let W_{opt} be the solution of the adaptive equalizer based on the cost function (3) that yields the correct signal constellation. All the weight vectors

$$W_x = \exp(j\phi)W_{opt}, \quad 0 \leq \phi \leq 2\pi \tag{6}$$

produce the same cost as $\bar{J}_{CMA}(W_{opt})$. Thus the CMA blind equalizer may converge to any of the solutions defined in equation (6). Since equation (6) produces the same amplitude over the interval $(0, 2\pi)$ which is also mentioned in [3], this undesired phase shift cannot be resolved by the CMA and must be eliminated by other methods.

2.2 Blind Channel Equalization with the Fuzzy Logic Based CMA (F-CMA) Algorithm

We present a novel approach where fuzzy logic can be used to adjust the step size for the CMA in this paper. The proposed approach is based on the principle of fuzzy logic developed by Zadeh [9], which is used to handle the linguistic concepts. It consists of three main processors, namely: **Fuzzifier**, **Rule Decision Table**, and **Defuzzifier**, which map the input variables into a suitable step size for the CMA weight adaptation. The block diagram of the proposed Fuzzy Logic based CMA (F-CMA) algorithm, which is used to obtain the step size appropriately in the CMA, is given in Figure 1.

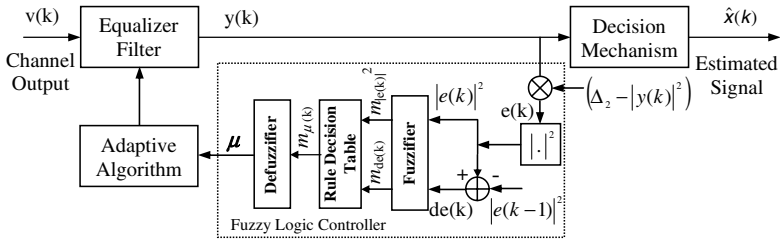


Fig. 1. The block diagram of fuzzy logic based CMA (F-CMA) algorithm

The fuzzy logic based controllers adjust the system input by observing the system output and by using the history of the information or criteria accumulated as an expert [10, 11]. For the system under investigation in this study, the power of the error $|e(k)|^2$ and the change of which $de(k) = |e(k)|^2 - |e(k-1)|^2$ are applied as the input signals to the fuzzy logic based controller as shown in Figure 1. Then they are used in the rule decision table to determine the fuzzy number of the output control signal. Finally the resultant fuzzy numbers representing the controller output are converted to the crisp values. In our application, the fuzzy variables are represented as linguistic variables by using three fuzzy sets for each input. These linguistic variables are defined for $|e(k)|^2$ and μ as positive small (PS), positive medium (PM) and positive big (PB) and for $de(k)$ these variables are defined as negative big (NB), zero (Z) and positive big (PB). These fuzzy sets have Gaussian transition forms which are known as membership functions $m_{|e(k)|^2}$, $m_{de(k)}$ and $m_{\mu(k)}$. The shapes of these fuzzy sets were chosen as Gaussian membership functions, and are given in Figure 2.

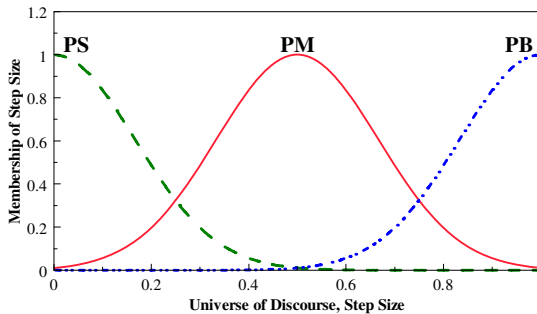


Fig. 2. The membership functions defined for the step size (μ)

Each of the membership functions can be analytically defined by the equation (7).

$$MF_G = \exp \left[- \frac{\left(|e(k)|^2 - |e(k)_p|^2 \right)^2}{2D} \right] \tag{7}$$

Where, $|e(k)|_p^2$ is the crisp number corresponding to the peak point of the Gaussian curve. D is the widening factor of the Gaussian function. However, the minimum and the maximum boundaries of the universes of discourses are different for three of them [10, 11, 12]. Since the magnitude of error, $e(k)$ varies between 0 and 2, the power of the error $|e(k)|^2$ may vary from 0 to 4, when a unit value is considered for the original transmit data. Therefore the minimum and the maximum limits of the universe of discourse for the power of the error, are taken as $|e(k)|_{\min}^2 = 0$ and $|e(k)|_{\max}^2 = 4$. On the other hand change in the power of the error is very small compared to $|e(k)|^2$. Therefore, the minimum and the maximum limits of $de(k)$ are assigned as $de(k)_{\min} = -|e(k)|_{\max}^2 / 20$ and $de(k)_{\max} = |e(k)|_{\max}^2 / 20$, in order to make the changes in power of the error more effective in terms of controller. Here, the divisor 20 is an arbitrary number referencing the maximum/minimum value of derivation to maximum value of error as in [12]. During the simulation analysis/studies the divisor can be chosen within 10 to 30, however 20 is a sensitive number produces better performance results. The minimum and the maximum values of the change in control signal (step size, μ) are 0 and 1, respectively as shown in Figure 2.

After the fuzzy numbers representing $|e(k)|^2$ and $de(k)$ are obtained, the fuzzy number of μ is determined using the assignments given in rule decision table. The rule decision table, which defines the control actions to be taken, is given in Table 1. Since there are three fuzzy membership functions for $|e(k)|^2$ and $de(k)$, a rule decision table with 9 rules are formed.

Table 1. The Rule Decision Table

$ e(k) ^2 \setminus de(k)$	NB _{de(k)}	Z _{de(k)}	PB _{de(k)}
PS _{e(k) ²}	PS _{μ} ¹	PS _{μ} ²	Z _{μ} ³
PM _{e(k) ²}	PS _{μ} ⁴	Z _{μ} ⁵	PM _{μ} ⁶
PB _{e(k) ²}	PM _{μ} ⁷	PS _{μ} ⁸	PS _{μ} ⁹

The fuzzy number representing the change in step size is determined from the rule table in a form of fuzzy linguistic terms. For example, if $|e(k)|^2$ has a non-zero membership degree in the fuzzy number PM and $de(k)$ has a non-zero membership degree in the fuzzy number NB, then the rule 4 is fired indicating that μ has a non-zero membership degree in the fuzzy number PS. This rule is expressed by a linguistic term as:

If $|e(k)|^2$ is PM and $de(k)$ is NB then μ is PS

This is called a rule. Multiple rules are connected to each other by the term ELSE. Therefore, the linguistic representation of the rule table given by Table 1 becomes as follows:

Rule 1: If $|e(k)|^2$ is PS and $de(k)$ is NB then μ is PS else

Rule 2: If $|e(k)|^2$ is PS and $de(k)$ is Z then μ is PS else

... ..

Rule 9: If $|e(k)|^2$ is PB and $de(k)$ is PB then μ is PS.

In the last stage, a reverse operation is performed to get the crisp number (μ) by the defuzzifier and the step size is determined by this way [10, 11, 13]. Different methods are applied to convert the fuzzy numbers to crisp equivalencies. The center of area (COA), which is also called the center of gravity method, is used in this study. The crisp value corresponding to the center of the resultant fuzzy subset areas are assigned for μ when the COA method is applied. This operation called COA method is defined in the following equation (8).

$$\mu = \frac{\sum_{k=1}^n m_{\mu}(k)\mu(k)}{\sum_{k=1}^n m_{\mu}(k)} \tag{8}$$

Where n is the number of the rules and $m_{\mu}(k)$ is the membership function value at location, $\mu(k)$. The reason for using the COA method instead of other defuzzification methods is because the COA method produces the smallest MSE.

Additional computational cost is incurred in the fuzzification (lookup table), inference (min, max operators) and defuzzification modules. There are also additional costs which come from the preparation of input variables prior to the fuzzification process (e.g. perform error squared). The computational complexity is defined as the floating point operation number which is required in processing the necessary data to find the desired weight vector in this study.

In the COA method used in the defuzzifying unit of the fuzzy logic controller maximum 4 multiplications, 1 division and 8 summations operation is performed [10, 11]. M in the table demonstrates the equalizer order. The comparison of the computational complexities of the equalizers required for per weight update is given in Table 2.

Table 2. Comparison of the Computational Complexities of the Equalizers

Equalizers	Multiplications	Divisions	Additions	Exp(.) Evaluations
CMA	8M + 6	-	8M	-
CMA-DD	16M + 8	-	20M	-
CMA-SDD	12M + 29	-	14M + 21	4
Proposed F-CMA	8M + 11	1	8M+8	2

The additional computational complexity brought by the fuzzy logic to the CMA is considerably low as can be seen in Table 2. It also can be seen that computational complexity per weight update of this proposed new scheme is simpler than that of the CMA, CMA-DD and CMA-SDD scheme.

3 Computer Simulation Results

The performance of the proposed low complexity F-CMA blind equalizer is compared with that of the N-CMA [1], M-CMA [2] and CMA-SDD [3] in computer simulations using the conventional CMA blind equalizer as a benchmark. The simulations are performed by using the loop of 100 Monte Carlo for the Quadrature Phase Shift Keying (QPSK) systems. Mean Square Error (MSE) performance criterion is used to assess the convergence rate of blind equalizers. Three channels (5 taps, 22 taps and 13 taps) are used in evaluation, and their channel impulse responses (CIR) are given in [3, 7], respectively.

In all the cases, a 23-tap Linear Transversal Equalizer (LTE) is used as in [3]. The equalizers are initialized with the central tap weight set to 1 and others set to 0. Table 3 lists the algorithm parameters used in the simulations for the four blind equalizers (μ for the CMA, μ_m and α_m for the M-CMA, α_n and σ_n for the N-CMA, and μ_c , μ_d and ρ for the CMA-SDD). All the simulations are obtained in the value of Signal to Noise Ratio (SNR) of 30 dB for the QPSK systems whose data sequence length is 4000. The comparison of the MSE convergence performances of the equalizers related to the blind channel equalization are given in Figure 3, 4 and 5.

Table 3. Algorithm Parameter Settings in Simulation

Simulations	CMA	M-CMA		N-CMA		CMA-SDD		
	μ	μ_m	α_m	α_n	σ_n	μ_c	μ_d	ρ
5 taps ch.	0.0018	0.05	0.01	0.5	0.009	0.0045	0.003	0.6
22 taps ch.	0.002	0.09	0.7	0.86	0.009	0.005	0.002	0.6
13 taps ch.	0.0025	0.088	0.85	0.7	0.009	0.0065	0.0025	0.7

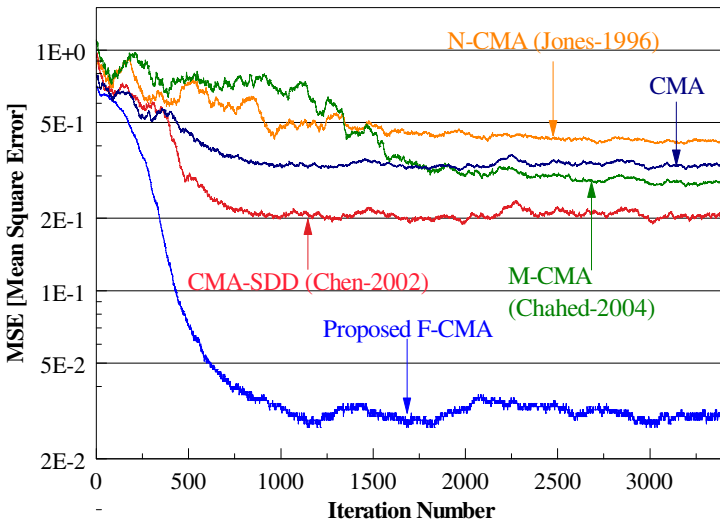


Fig. 3. Comparison of the MSE convergence performances for 5 tap CIR [3]

The learning curves of the five blind equalizers are compared in Figure 3. As can be seen from the figure above the MSE performance obtained with the N-CMA is a little worse than the CMA. The M-CMA passes the performance of the CMA by about 2000 step number and converges to the lower MSE floor. The CMA-SDD algorithm converges to the lower MSE floor by speeding up the CMA. However, the proposed

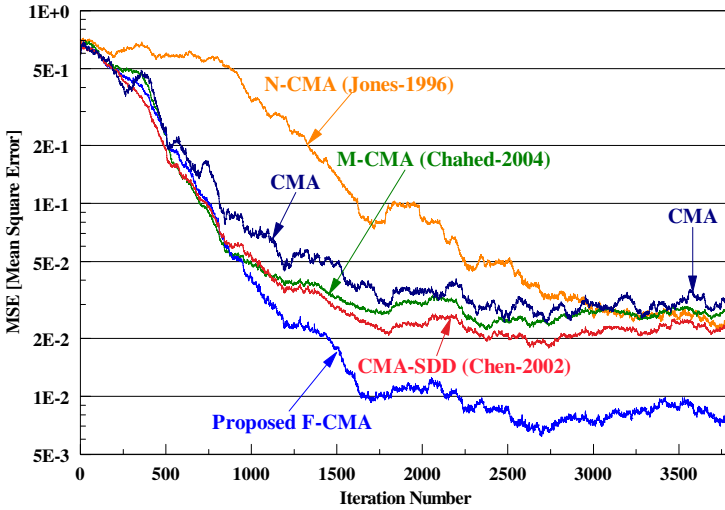


Fig. 4. Comparison of the MSE convergence performances for 22 tap CIR [3]

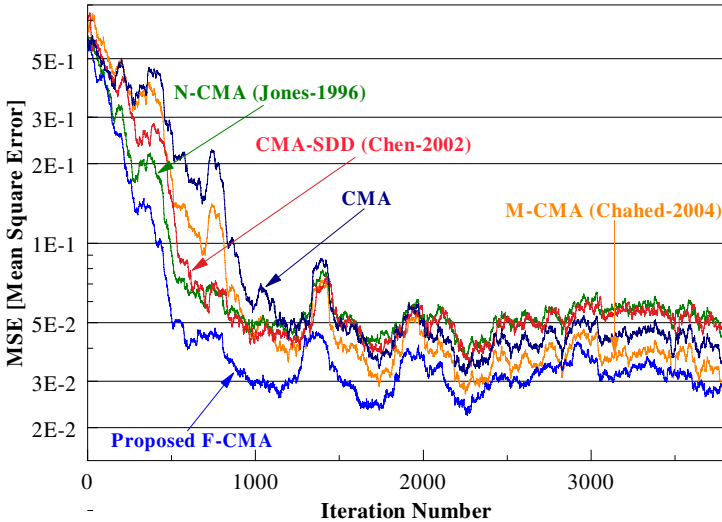


Fig. 5. Comparison of the MSE convergence performances for 13 tap CIR [7]

F-CMA developed in this study by accelerating the CMA perfectly has passed the CMA-SDD and other CMAs and has converged to the lowest steady state MSE floor.

The MSE performance obtained with the N-CMA passes the CMA just after 3000 step number as can be seen from the Figure 4. The M-CMA speeds up the performance of the CMA initially but converges to the nearly performance of the CMA algorithm. The CMA-SDD algorithm by speeding up the CMA converges to the lower MSE floor. However, the proposed F-CMA by accelerating the CMA has passed the CMA-SDD and other CMAs above mentioned and have converged to the lowest steady state MSE floor.

As can be seen from the Figure 5 the MSE performance obtained with the N-CMA algorithm accelerates the CMA initially but converges poorer than the CMA after 1000 step number. The M-CMA algorithm by accelerating the performance of the CMA provides performance significantly. The CMA-SDD algorithm accelerates the CMA initially but converges a little worse than the CMA. However, the proposed F-CMA algorithm by accelerating the CMA has passed the CMA algorithms above mentioned and has converged to the lowest steady state MSE floor.

Figure 6 shows the MSE versus SNR curves for all considered algorithms using the five taps channel profile given by [3] and the MSE performance samples are obtained after 4000 iterations of blind trainings. For an ordinary, training based equalizations bit-error-rate versus SNR curves are more common for performance comparisons, however considered blind trainings has no information about the incoming phase of transmit data and there is no way to detect the phase ambiguity error without any aid from data [3][6]. So, the produced MSE versus SNR curves exhibit a comparative error performance profile between algorithms like BER performance comparison curves except it is impossible to clear the MSE floor even for great values of SNR.

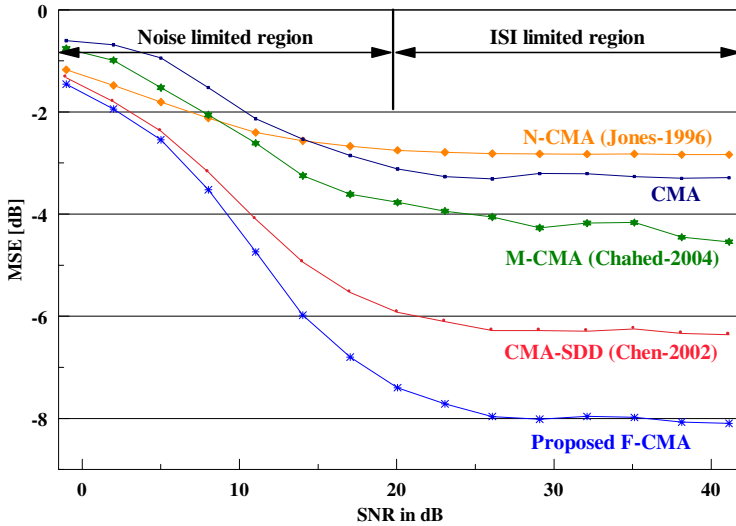


Fig. 6. Comparison of the MSE versus SNR performances for 5 tap CIR [3]

From Figure 6, the proposed F-CMA algorithm produced better steady state MSE performances in both SNR regions, noise-limited and ISI-limited. In essence, the proposed method aims to control the training due to error level and trend of error. Thus, any error comes from either ISI or noise would be taken into account and the proposed method obtains an efficient training which would be robust to whatever the error source.

4 Conclusion

In this paper, a novel low complexity blind equalization scheme based on fuzzy logic outer loop controller has been proposed. It has been shown that a combination of fuzzy logic and CMA equalizer provides an effective and robust way for adaptive blind equalization. Compared with a state of art low complexity blind equalization schemes, namely the recently introduced N-CMA [1], M-CMA [2] and CMA-SDD [3], the proposed F-CMA blind equalizer has simpler computational requirements, faster convergence and lower steady state equalization performance. This new blind equalizer offers practical alternatives to blind equalization of QPSK channels and provides significant MSE performance improvement over the conventional CMA blind equalizer. The results also have demonstrated that the fuzzy rules are able to adjust the step size effectively in order to obtain a faster and more stable training with the CMA and help to determine the convergence feature of a system.

References

1. Jones, D.L.: A Normalized Constant Modulus Algorithm. In: IEEE Conference Record of the Twenty-Ninth Asilomar Conference on Signals, Systems and Computers, vol. 1, pp. 694–697 (1996)
2. Chahed, I., et al.: Blind Decision Feedback Equalizer Based on High Order MCMA. In: Canadian Conf. on Electrical and Computer Engineering, vol. 4, pp. 2111–2114 (2004)
3. Chen, S., et al.: Low Complexity Concurrent Constant Modulus Algorithm and Soft Decision Directed Scheme for Blind Equalization. IEE Proc. Visual Image Signal Proces. 150, 312–320 (2003)
4. Godard, D.N.: Self Recovering Equalization and Carrier Tracking in Two Dimensional Data Communication Systems. IEEE Trans. Comm. 28(11), 1867–1875 (1980)
5. Treichler, J.R., Agee, B.G.: A New Approach to Multipath Correction of Constant Modulus Signals. IEEE Trans. Acoust. Speech, Signal Process. ASSP-28, 459–472 (1983)
6. Schirtzinger, T., et al.: A Comparison of Three Algorithms for Blind Equalization Based on the Constant Modulus Error Criterion. In: Proc. IEEE ICASSP, pp. 1049–1052 (1985)
7. Weerackody, V., Kassam, S.A.: Variable Step Size Blind Adaptive Equalization Algorithms. In: IEEE International Symposium on Circuits and Systems, pp. 718–721 (1991)
8. Castro, D.F.C.C., et al.: Concurrent Blind Deconvolution for Channel Equalization. In: Proc. ICC, vol. II, pp. 366–371 (2001)
9. Zadeh, L.A.: Fuzzy Sets. Information and Con. 8, 338–353 (1965)
10. Özen, A.: A Fuzzy Based Outer Loop Controller Design Improving the Performance and Convergence Speed in High Data Rate Digital Communication Receivers. Ph.D. Thesis, Graduate School of Natural and Applied Science, KTU, Trabzon (2005)

11. Özen, A., Kaya, I., Soysal, B.: Design of a Fuzzy Based Outer Loop Controller for Improving the Training Performance of LMS Algorithm. In: Huang, D.-S., Heutte, L., Loog, M. (eds.) ICIC 2007. CCIS, vol. 2, pp. 1051–1063. Springer, Heidelberg (2007)
12. Eminoğlu, İ., Altaş, İ.H.: A Method to Form Fuzzy Logic Control Rules for a PMDC Motor Drive System. Elsevier Electric Power System Research 39, 81–87 (1996)
13. Özen, A., et al.: A Fuzzy-based Outer Loop Controller for LMS Algorithm and Its Application to Channel Estimation and Carrier Tracking for OFDM. In: MOMUC 2003 8th International Workshop on Mobile Multimedia Communications, pp. 79–86 (2003)

A Risk-Sensitive Portfolio with Mean and Variance of Fuzzy Random Variables

Yuji Yoshida

Faculty of Economics and Business Administration, University of Kitakyushu
4-2-1 Kitagata, Kokuraminami, Kitakyushu 802-8577, Japan
yoshida@kitakyu-u.ac.jp

Abstract. This paper discusses a risk-sensitive portfolio problem, where the objective function is defined by randomness and fuzziness, and it introduces the perception-based extension of the expectation and the variance for fuzzy random variables. Fuzzy random variables are estimated by mean and variance with λ -mean functions and evaluation weights: A possibility-necessity weight ν for subjective estimation, and a pessimistic-optimistic index λ for subjective decision. A solution of the risk-sensitive portfolio problem is derived by quadratic programming approach.

1 Introduction

Fuzzy random variables, which were introduced by Kwakernaak [7] are applied to decision-making under uncertainty with fuzziness like linguistic data in statistics, engineering and economics. The notion of fuzzy random variable is strongly related to perception of random events. The relation between fuzzy logic with randomness and perception is found in Zadeh [15]. On the other hand, in decision-making with randomness we need estimations of real random variables like the expectation as a criterion in optimization. This paper discusses perception-based extension of means and variances for fuzzy random variables. In this paper, a set of perceived random events, which are represented by real random variables, is given by a fuzzy random variable. An estimation of perceived random events is also represented by a functional on real random variables. Then, we obtain the estimation of the perception regarding random events, extending the functional to a functional of fuzzy random variables.

The portfolio is one of the most important tools for the asset management in finance. Portfolio models have been studied by many authors using mathematical programming on the basis of Markowitz's model ([8,9]). When we deal with systems like financial markets, fuzzy logic works well since the markets contain the uncertain factors which are different from probabilistic essence and in which it is difficult to specify actual price values exactly ([5]). In this paper, randomness is applied to the uncertainty regarding the belief degree of frequency, and fuzziness is applied to imprecision of data because of a lack of knowledge regarding the current stock market. In this paper, we consider a risk-sensitive portfolio model under uncertainty of randomness and fuzziness.

Estimation of uncertain quantities is important in decision making. To represent uncertainty in a finance model, we use *fuzzy random variables* which have

two kinds of uncertainties, i.e. randomness and fuzziness. Recently Yoshida [11,12] introduced mean, variance and covariance of fuzzy random variables using *evaluation weights* and *λ -mean functions*. In this paper, we estimate fuzzy random variables by probabilistic expectation and these evaluations, which are characterized by a *possibility-necessity weight* for subjective estimation and a *pessimistic-optimistic index* for subjective decision. In a portfolio model, we use triangle-type fuzzy numbers/fuzzy random variables for simplicity in numerical computation when we apply them to actual models. We discuss a risk-sensitive problem for the portfolio, where the risk is defined by both random factors and fuzzy factors in the model. By quadratic programming approach, we derive a solution of the risk-sensitive portfolio problem.

2 Perception-Based Extension of Estimations

First we introduce some notations of fuzzy numbers. Let \mathbb{R} denote the set of all real numbers. A fuzzy number is denoted by its membership function $\tilde{a} : \mathbb{R} \mapsto [0, 1]$ which is normal, upper-semicontinuous, fuzzy convex and has a compact support ([6]). \mathcal{R} denotes the set of all fuzzy numbers. In this paper, we identify fuzzy numbers with their corresponding membership functions ([14]). The α -cut of a fuzzy number \tilde{a} is given by closed intervals $\tilde{a}_\alpha := \{x \in \mathbb{R} \mid \tilde{a}(x) \geq \alpha\}$ ($\alpha \in (0, 1]$) and $\tilde{a}_0 := \text{cl}\{x \in \mathbb{R} \mid \tilde{a}(x) > 0\}$, where cl denotes the closure of an interval and $:=$ means that the left term is defined by the right term. The α -cut is written by closed intervals $\tilde{a}_\alpha = [\tilde{a}_\alpha^-, \tilde{a}_\alpha^+]$ ($\alpha \in [0, 1]$). Hence we introduce a partial order \succeq , so called the *fuzzy max order*, on fuzzy numbers \mathcal{R} ([6]): For $\tilde{a}, \tilde{b} \in \mathcal{R}$,

$$\tilde{a} \succeq \tilde{b} \quad \text{means that} \quad \tilde{a}_\alpha^- \geq \tilde{b}_\alpha^- \quad \text{and} \quad \tilde{a}_\alpha^+ \geq \tilde{b}_\alpha^+ \quad \text{for all } \alpha \in [0, 1].$$

An addition, a subtraction and a scalar multiplication for fuzzy numbers are defined as follows ([14]): Let $\tilde{a}, \tilde{b} \in \mathcal{R}$ and $\zeta \geq 0$. The addition and subtraction $\tilde{a} \pm \tilde{b}$ of \tilde{a} and \tilde{b} and the scalar multiplication $\zeta\tilde{a}$ of ζ and \tilde{a} are fuzzy numbers given respectively by their α -cuts as follows

$$(\tilde{a} \pm \tilde{b})_\alpha := [\tilde{a}_\alpha^- \pm \tilde{b}_\alpha^-, \tilde{a}_\alpha^+ \pm \tilde{b}_\alpha^+] \quad \text{and} \quad (\zeta\tilde{a})_\alpha := [\zeta\tilde{a}_\alpha^-, \zeta\tilde{a}_\alpha^+]$$

for α -cuts $\tilde{a}_\alpha = [\tilde{a}_\alpha^-, \tilde{a}_\alpha^+]$ and $\tilde{b}_\alpha = [\tilde{b}_\alpha^-, \tilde{b}_\alpha^+]$ ($\alpha \in [0, 1]$).

Let P be a non-atomic probability on a sample Ω . Let \mathcal{X} be the set of all integrable real random variables on (Ω, P) . A fuzzy-number-valued map $\tilde{X} : \Omega \mapsto \mathcal{R}$ is called a *fuzzy random variable* if the maps $\omega \mapsto \tilde{X}_\alpha^\pm(\omega)$ are measurable for all $\alpha \in (0, 1]$, where

$$\tilde{X}_\alpha(\omega) := [\tilde{X}_\alpha^-(\omega), \tilde{X}_\alpha^+(\omega)] := \{x \in \mathbb{R} \mid \tilde{X}(\omega)(x) \geq \alpha\}$$

(see [7,10]). A fuzzy random variable \tilde{X} is said to be *integrable* if $\omega \mapsto \tilde{X}_\alpha^\pm(\omega)$ are integrable for all $\alpha \in (0, 1]$. The *expectation* of an integrable fuzzy random variable \tilde{X} is a fuzzy number

$$\tilde{E}(\tilde{X})(x) := \sup_{X \in \mathcal{X} : E(X)=x} \inf_{\omega \in \Omega} \tilde{X}(\omega)(X(\omega)), \quad x \in \mathbb{R}, \quad (1)$$

where \mathcal{X} is taken as the set of all integrable real random variables and $E(X) := \int X dP$ (Kruse and Meyer [4]). Then, it is known that the expectation $\tilde{E}(\tilde{X})$ is a fuzzy number whose α -cut is given by

$$\tilde{E}(\tilde{X})_\alpha = [E(\tilde{X}_\alpha^-), E(\tilde{X}_\alpha^+)] \tag{2}$$

for $\alpha \in (0, 1]$. This extension is well-defined and has monotone, continuous and linear properties in Yoshida [13]. On the other hand, the variance of a real random variable X is given by

$$V(X) = E((X - E(X))^2) = \int (X - E(X))^2 dP. \tag{3}$$

Hence we can give a definition of the extended estimation $\tilde{V}(\tilde{X})$ by

$$\tilde{V}(\tilde{X})(x) := \sup_{X \in \mathcal{X} : V(X)=x} \inf_{\omega \in \Omega} \tilde{X}(\omega)(X(\omega)), \quad x \in \mathbb{R}, \tag{4}$$

and this definition implies that its α -cut is

$$\{E((X - E(X))^2) | X \in \mathcal{X}, X(\omega) \in \tilde{X}_\alpha(\omega) \text{ for all } \omega \in \Omega\}. \tag{5}$$

Then, *the variance does not satisfy the monotone properties* in Yoshida [13]. In general, this set does not equal to the form of the interval (2) since the monotone properties does not hold. Thus the extension of variance is not well-defined. We can find some approaches regarding the variance in Carlsson and Fullér [1], Feng et al. [2] and Yoshida [12]. In this paper, we introduce the variance with λ -mean functions and evaluation weights so that the extended variances are compatible to the extended means.

3 Mean and Variance for Fuzzy Random Variables

There are many research results for the estimation of fuzzy numbers ([3]). In this paper, we use an evaluation method of fuzzy numbers/fuzzy random variables introduced by Yoshida [11] to estimate the rate of return in the portfolio. They studied an evaluation of fuzzy numbers by *evaluation weights* which are induced from fuzzy measures to evaluate a confidence degree that a fuzzy number takes values in an interval. With respect to fuzzy random variables, the randomness is evaluated by the probabilistic expectation and the fuzziness is estimated by evaluation weights and the following function. Let $g^\lambda : \mathcal{I} \mapsto \mathbb{R}$ be a map such that

$$g^\lambda([x, y]) := \lambda x + (1 - \lambda)y, \quad [x, y] \in \mathcal{I}, \tag{6}$$

where λ is a constant satisfying $0 \leq \lambda \leq 1$ and \mathcal{I} denotes the set of all bounded closed intervals. This scalarization is used for the estimation of fuzzy numbers to give a mean value of the interval $[x, y]$ with a weight λ , and λ is called a *pessimistic-optimistic index* and indicates the pessimistic degree in decision making ([3]). Then, g^λ is called a λ -mean function. Let a fuzzy number $\tilde{a} \in \mathcal{R}$.

A mean value of the fuzzy number \tilde{a} with respect to λ -mean functions g^λ and an evaluation weights $w(\alpha)$, which is derived from fuzzy measures, is given by ([11][12])

$$\tilde{E}^\lambda(\tilde{a}) := \int_0^1 g^\lambda(\tilde{a}_\alpha) w(\alpha) d\alpha \Big/ \int_0^1 w(\alpha) d\alpha, \tag{7}$$

where $\tilde{a}_\alpha = [\tilde{a}_\alpha^-, \tilde{a}_\alpha^+]$ is the α -cut of the fuzzy number \tilde{a} . In (7), $w(\alpha)$ indicates a confidence degree that the fuzzy number \tilde{a} takes values in the interval \tilde{a}_α at each level α . Hence, an evaluation weight $w(\alpha)$ is called the possibility evaluation weight $w^P(\alpha)$ and the necessity evaluation weight $w^N(\alpha)$ induced from the fuzzy number \tilde{a} if they are given respectively by $w^P(\alpha) = 1$ and $w^N(\alpha) = 1 - \alpha$ for $\alpha \in [0, 1]$. Especially, for a fuzzy number $\tilde{a} \in \mathcal{R}$, the mean $\tilde{E}^P(\tilde{a})$ in the possibility case and the mean $\tilde{E}^N(\tilde{a})$ in the necessity case are represented as follows ([11][12]): $\tilde{E}^P(\tilde{a}) = \int_0^1 g^\lambda(\tilde{a}_\alpha) d\alpha$ and $\tilde{E}^N(\tilde{a}) = \int_0^1 g^\lambda(\tilde{a}_\alpha) (2 - 2\alpha) d\alpha$. The mean \tilde{E}^λ has natural properties regarding the linearity and the monotonicity regarding the fuzzy max order ([11][12]). Using evaluation weights, we give means, variances and covariances regarding fuzzy random variables. For a fuzzy random variable \tilde{X} , the mean of the expectation $E(\tilde{X})$ is a real number

$$E(\tilde{E}^\lambda(\tilde{X})) = E \left(\int_0^1 g^\lambda(\tilde{X}_\alpha) w(\alpha) d\alpha \Big/ \int_0^1 w(\alpha) d\alpha \right). \tag{8}$$

From [11][12], we obtain the following results regarding fuzzy random variables.

Lemma 1. Let $\lambda \in [0, 1]$. For a fuzzy number $\tilde{a} \in \mathcal{R}$, integrable fuzzy random variables \tilde{X}, \tilde{Y} and an integrable real random variable Z and a nonnegative real number ζ , the following (i) – (vi) hold.

- (i) $E(\tilde{E}^\lambda(\tilde{X})) = \tilde{E}^\lambda(E(\tilde{X}))$.
- (ii) $E(\tilde{E}^\lambda(\tilde{a})) = \tilde{E}^\lambda(\tilde{a})$ and $E(\tilde{E}^\lambda(Z)) = E(Z)$.
- (iii) $E(\tilde{E}^\lambda(\tilde{X} + \tilde{a})) = E(\tilde{E}^\lambda(\tilde{X})) + \tilde{E}^\lambda(\tilde{a})$ and $E(\tilde{E}^\lambda(\tilde{X} + Z)) = E(\tilde{E}^\lambda(\tilde{X})) + E(Z)$.
- (iv) $E(\tilde{E}^\lambda(\zeta\tilde{X})) = \zeta E(\tilde{E}^\lambda(\tilde{X}))$.
- (v) $E(\tilde{E}^\lambda(\tilde{X} + \tilde{Y})) = E(\tilde{E}^\lambda(\tilde{X})) + E(\tilde{E}^\lambda(\tilde{Y}))$.
- (vi) If $\tilde{X} \succeq \tilde{Y}$, then $E(\tilde{E}^\lambda(\tilde{X})) \geq E(\tilde{E}^\lambda(\tilde{Y}))$, where \succeq is the fuzzy max order.

Let $\tilde{a} \in \mathcal{R}$ be a fuzzy number and let $\nu \in [0, 1]$. For applications of the mean values in actual problems, we can introduce a mean value with a parameter ν :

$$\tilde{E}^{\lambda,\nu}(\tilde{a}) := \nu\tilde{E}^P(\tilde{a}) + (1 - \nu)\tilde{E}^N(\tilde{a}). \tag{9}$$

Then, ν is called a possibility-necessity weight, and (9) is the mean with the possibility-necessity weight ν . From [11][12], the mean $\tilde{E}^{\lambda,\nu}(\cdot)$ is represented by

$$\tilde{E}^{\lambda,\nu}(\tilde{a}) = \int_0^1 g(\tilde{a}_\alpha) (\nu + 2(1 - \nu)(1 - \alpha)) d\alpha. \tag{10}$$

Next we introduce variances and covariances of fuzzy random variables from the viewpoint of λ -mean functions and evaluation weights. For fuzzy random variables \tilde{X} and \tilde{Y} , we define variances and covariances as follows ([12]).

$$V^\lambda(\tilde{X}) := E \left(\frac{\int_0^1 (g^\lambda(\tilde{X}_\alpha) - E(g^\lambda(\tilde{X}_\alpha)))^2 w(\alpha) d\alpha}{\int_0^1 w(\alpha) d\alpha} \right),$$

$$Cov^{\lambda,\gamma}(\tilde{X}, \tilde{Y}) := E \left(\frac{\int_0^1 (g^\lambda(\tilde{X}_\alpha) - E(g^\lambda(\tilde{X}_\alpha)))(g^\gamma(\tilde{Y}_\alpha) - E(g^\gamma(\tilde{Y}_\alpha))) w(\alpha) d\alpha}{\int_0^1 w(\alpha) d\alpha} \right)$$

for $\lambda, \gamma \in [0, 1]$. We can find other approaches in [12] which discuss the variance of fuzzy numbers by possibility theory. Hence we obtain the following natural properties about the variance $V^\lambda(\cdot)$ and covariance $Cov^{\lambda,\gamma}(\cdot, \cdot)$.

Lemma 2. *Let $\lambda, \gamma \in [0, 1]$. For fuzzy numbers $\tilde{a}, \tilde{b} \in \mathcal{R}$, integrable fuzzy random variables \tilde{X}, \tilde{Y} , integrable real random variables X, Y and a nonnegative real number ζ , the following (i) – (v) hold.*

- (i) $V^\lambda(\tilde{a}) = 0$ and $V^\lambda(X) = V(X)$, where $V(\cdot)$ is the variance of real r.v..
- (ii) $V^\lambda(\tilde{X} + \tilde{a}) = V^\lambda(\tilde{X})$.
- (iii) $V^\lambda(\zeta\tilde{X}) = \zeta^2 V^\lambda(\tilde{X})$.
- (iv) $Cov^{\lambda,\gamma}(\tilde{X}, \tilde{a}) = Cov^{\lambda,\gamma}(\tilde{a}, \tilde{X}) = 0$ and $Cov^{\lambda,\gamma}(X, Y) = Cov(X, Y)$, where $Cov(\cdot, \cdot)$ is the covariance of real random variables.
- (v) $Cov^{\lambda,\gamma}(\tilde{X} + \tilde{a}, \tilde{Y} + \tilde{b}) = Cov^{\lambda,\gamma}(\tilde{X}, \tilde{Y})$.

4 A Risk-Sensitive Portfolio Model under Uncertainty

In this paper, we consider a portfolio model with a bond and n stocks, where n is a positive integer. In the remainder of this section, we describe a bond price process and stock price processes. We deal with a model where an investor’s actions do not have any impact on the stock market, so-called *small investors hypothesis* [9]. Let $\mathbb{T} := \{0, 1, 2, \dots, T\}$ be the time space with an expiration date T , and \mathbb{R} denotes the set of all real numbers. Take a probability space $\Omega := (\mathbb{R}^n)^{T+1}$ by the product of \mathbb{R} and let P be a non-atomic probability measure on Ω . For an asset $i = 1, 2, \dots, n$, a *stock price process* $\{S_t^i\}_{t=0}^T$ is given by *rates of return* R_t^i as follows: $S_t^i := S_{t-1}^i(1 + R_t^i)$ for $t = 1, 2, \dots, T$, where $\{R_t^i\}_{t=1}^T$ is assumed to be a sequence of integrable independent real random variables. Then we have

$$S_t^i = S_0^i \prod_{s=1}^t (1 + R_s^i) \tag{11}$$

for $t = 1, 2, \dots, T$. In this paper, we discuss a portfolio model where stock price processes S_t^i take fuzzy values using fuzzy random variables. Now we deal with a case when the rate of return $\{R_t^i\}_{t=1}^T$ has some imprecision. We define a *rate of return process with imprecision* $\{\tilde{R}_t^i\}_{t=0}^T$ by a sequence of triangle-type fuzzy random variables

$$\tilde{R}_t^i(\cdot)(x) = \begin{cases} \frac{x - R_t^i + c_t^i}{c_t^i} & \text{if } R_t^i - c_t^i \leq x < R_t^i \\ \frac{x - R_t^i - c_t^i}{-c_t^i} & \text{if } R_t^i \leq x < R_t^i + c_t^i \\ 0 & \text{otherwise,} \end{cases} \tag{12}$$

where c_t^i is a positive number. We call c_t^i a *fuzzy factor* for asset i at time t . Hence we can represent \tilde{R}_t^i by the sum of fuzzy numbers:

$$\tilde{R}_t^i(\omega)(\cdot) := 1_{\{R_t^i(\omega)\}}(\cdot) + \tilde{a}_t^i(\cdot) \tag{13}$$

for $\omega \in \Omega$, where $1_{\{\cdot\}}$ denotes the characteristic function of a singleton and \tilde{a}_t^i is a fuzzy number defined by $\tilde{a}_t^i(\cdot) := \max\{(1 - |\cdot|/c_t^i), 0\}$. For assets $i = 1, 2, \dots, n$, we define *stock price processes* $\{\tilde{S}_t^i\}_{t=0}^T$ by the *rates of return with imprecision* \tilde{R}_t^i as follows: $\tilde{S}_0^i := S_0^i$ is a constant and

$$\tilde{S}_t^i = \tilde{S}_0^i \prod_{s=1}^t (1 + \tilde{R}_s^i) \tag{14}$$

for $t = 1, 2, \dots, T$. Hence, we deal with a portfolio with *portfolios given by portfolio weight vectors* $w = (w^1, w^2, \dots, w^n)$ such that $w^1 + w^2 + \dots + w^n = 1$ and $w^i \geq 0$ ($i = 1, 2, \dots, n$). For the portfolio $w = (w^1, w^2, \dots, w^n)$, the rate of return with imprecision for the portfolio is given by

$$\tilde{R}_t := w^1 \tilde{R}_t^1 + w^2 \tilde{R}_t^2 + \dots + w^n \tilde{R}_t^n. \tag{15}$$

In this paper, we discuss a risk-sensitive model regarding (15) under guarantee of rewards. Let the mean, variance and covariance of the fuzzy random variables \tilde{R}_t^i by $\tilde{\mu}_t^i := E(\tilde{E}^\lambda(\tilde{R}_t^i))$, $(\tilde{\sigma}_t^i)^2 := V^\lambda(\tilde{R}_t^i)$ and $\tilde{\sigma}_t^{ij} := Cov^{\lambda, \lambda}(\tilde{R}_t^i, \tilde{R}_t^j)$ for $\lambda \in [0, 1]$ and $i, j = 1, 2, \dots, n$. From Lemmas 1 and 2 and (13), we obtain the following results regarding the rates of returns \tilde{R}_t^i : $\tilde{\mu}_t^i = \mu_t^i + \tilde{E}^\lambda(\tilde{a}_t^i)$, $(\tilde{\sigma}_t^i)^2 = (\sigma_t^i)^2$ and $\tilde{\sigma}_t^{ij} = \sigma_t^{ij}$, where $\mu_t^i := E(R_t^i)$, $(\sigma_t^i)^2 := V(R_t^i) = E((R_t^i - \mu_t^i)^2)$, $\sigma_t^{ij} := Cov(R_t^i, R_t^j) = E((R_t^i - \mu_t^i)(R_t^j - \mu_t^j))$. For the portfolio $w = (w^1, w^2, \dots, w^n)$, from Lemmas 1 and 2 the expectation $\tilde{\mu}_t$ and variance $(\tilde{\sigma}_t)^2$ regarding the *rate of return with imprecision for the portfolio* \tilde{R}_t in (15) is given as follows.

$$\tilde{\mu}_t := \sum_{i=1}^n w^i \tilde{\mu}_t^i = \sum_{i=1}^n w^i (\mu_t^i + \tilde{E}^\lambda(\tilde{a}_t^i)) \quad \text{and} \quad (\tilde{\sigma}_t)^2 := \sum_{i=1}^n \sum_{j=1}^n w^i w^j \sigma_t^{ij},$$

where $\sigma_t^{ii} = (\sigma_t^i)^2$. Hence, to guarantee the lower bound regarding the expectation $\tilde{\mu}_t$ of the rate of return for the portfolio we estimate $\tilde{\mu}_t$ taking the index λ pessimistic ($\lambda = 1$) and the necessity mean $w^N(\alpha) = 1 - \alpha$ ($\alpha \in [0, 1]$): For a positive constant κ , we have

$$\tilde{\mu}_t - \frac{\kappa}{2} (\tilde{\sigma}_t)^2 \geq \sum_{i=1}^n w^i \left(\mu_t^i - \frac{2}{3} c_t^i \right) - \frac{\kappa}{2} \sum_{i=1}^n \sum_{j=1}^n w^i w^j \sigma_t^{ij}. \tag{16}$$

for $\lambda \in [0, 1]$, where $\tilde{a}_\alpha^i = [\tilde{a}_\alpha^{i-}, \tilde{a}_\alpha^{i+}] = [-c_t^i(1 - \alpha), c_t^i(1 - \alpha)]$ from (14). Then we discuss the following risk-sensitive problem regarding the lower bound (16).

Risk-sensitive problem (RS): Maximize the risk-sensitive rate of return

$$\tilde{\rho} := \sum_{i=1}^n w^i \left(\mu_t^i - \frac{2}{3} c_t^i \right) - \frac{\kappa}{2} \sum_{i=1}^n \sum_{j=1}^n w^i w^j \sigma_t^{ij}. \tag{17}$$

with portfolios $w = (w^1, w^2, \dots, w^n)$ satisfying $w^1 + w^2 + \dots + w^n = 1$ and $w^i \geq 0 (i = 1, 2, \dots, n)$.

Theorem 1. *Assume $A > 0$ and $\tilde{\Sigma}^{-1}(A\tilde{\mu} + (\kappa - B)\mathbf{1}) \geq \mathbf{0}$. Then there exists a solution $\tilde{\rho}^*$ of Problem (RS), which is given by the following $\tilde{\rho}^*$:*

$$\tilde{\rho}^* := \frac{\Delta + 2\kappa B - \kappa^2}{2\kappa A} \tag{18}$$

with the expected rate of return

$$\tilde{\mu}^* = \frac{\Delta + \kappa B}{\kappa A}. \tag{19}$$

The corresponding portfolio w^* is given by

$$w^* = \frac{1}{\kappa} \left(\tilde{\Sigma}^{-1}\tilde{\mu} + \frac{\kappa - B}{A}\tilde{\Sigma}^{-1}\mathbf{1} \right), \tag{20}$$

where $\tilde{\sigma}_t^{ij} := \sigma_t^{ij}$ for $i, j = 1, 2, \dots, n$,

$$\tilde{\mu} := \begin{bmatrix} \mu_t^1 \\ \mu_t^2 \\ \vdots \\ \mu_t^n \end{bmatrix} - \frac{2}{3} \begin{bmatrix} c_t^1 \\ c_t^2 \\ \vdots \\ c_t^n \end{bmatrix}, \quad \tilde{\Sigma} := \begin{bmatrix} \tilde{\sigma}_t^{11} & \tilde{\sigma}_t^{12} & \dots & \tilde{\sigma}_t^{1n} \\ \tilde{\sigma}_t^{21} & \tilde{\sigma}_t^{22} & \dots & \tilde{\sigma}_t^{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{\sigma}_t^{n1} & \tilde{\sigma}_t^{n2} & \dots & \tilde{\sigma}_t^{nn} \end{bmatrix}, \quad \mathbf{1} := \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix},$$

$A := \mathbf{1}^T \tilde{\Sigma}^{-1} \mathbf{1}$, $B := \mathbf{1}^T \tilde{\Sigma}^{-1} \tilde{\mu}$, $C := \tilde{\mu}^T \tilde{\Sigma}^{-1} \tilde{\mu}$, $\Delta := AC - B^2$ and $\mathbf{0}$ is zero vector.

5 A Numerical Example and Concluding Remarks

We can deal with a combination of the possibility-necessity weight ν and the pessimistic-optimistic index λ . The decision maker may choose the parameters $\lambda (\in [0, 1])$ and $\nu (\in [0, 1])$. The pessimistic-optimistic index is taken as $\lambda = 1$ if he has pessimistic personal forecast in the market and he takes careful decision, and $\lambda = 0$ if he has optimistic personal forecast and he is not nervous. The possibility-necessity weight is taken as $\nu = 1$ when he has enough confidential information about the market, and $\nu = 0$ when he does not have confidential information. In this model, $\nu = 0$ is reasonable since our objective function is risk-sensitive, and we need to take into account of the fuzziness of information in the market. While λ depends on the decision maker's attitude in his investment. In this example, we compute the pessimistic and necessity case $\lambda = 1$ and $\nu = 0$. Let $n = 3$ and $\kappa = 1$. Give a mean and variance-covariance matrix of rate of return and fuzzy factors as follows: Expected rates of return $\mu_t^1 = 0.05, \mu_t^2 = 0.07, \mu_t^3 = 0.06$, variance-covariances $(\sigma_t^1)^2 = 0.40, (\sigma_t^2)^2 = 0.20, (\sigma_t^3)^2 = 0.30, \sigma_t^{12} = \sigma_t^{21} = 0.03, \sigma_t^{13} = \sigma_t^{31} = 0.02, \sigma_t^{23} = \sigma_t^{32} = -0.06$, and fuzzy factors $c_t^1 = 0.01, c_t^2 = 0.01, c_t^3 = 0.01$. Then we can easily check the assumption of Theorem 1, and we obtain the optimal portfolio $w^* = (w^1, w^2, w^3) = (0.116538, 0.521851, 0.361611)$. Table 1

Table 1. The rate of return, the risk sensitive rate and Sharpe ratio

	With fuzziness	Without fuzziness	Variance-mini
The rate of return $\tilde{\mu}^*$	0.04898	0.07001	0.04867
The risk sensitive rate $\tilde{\rho}^*$	0.00807	0.02858	0.03206
Sharpe ratio	0.03140	0.10445	0.02876

shows the expected rate $\tilde{\mu}^*$ with/without imprecision of data, the risk-sensitive rate $\tilde{\rho}^*$ and Sharpe ratio, which is a famous performance index of portfolios ([9]), for a rate of the market $r_t = 0.04$. This risk sensitive model looks better than a variance-minimizing model with fuzziness from Table 1.

This paper discusses a *risk-sensitive portfolio problem*, where the objective function is defined by randomness and fuzziness, and it is estimated with the *perception-based extension of the expectation and the variance for fuzzy random variables*. In Table 1, we can observe the loss from the fuzziness however it is not so serious for portfolio management because we can utilize *possibility-necessity weights* ν and *pessimistic-optimistic induces* λ in actual decision making. We hope the results in this paper are applied to various fields with uncertainty.

References

1. Carlsson, C., Fullér, R.: On Possibilistic Mean Value and Variance of Fuzzy Numbers. *Fuzzy Sets and Systems*, pp. 315–326 (2001)
2. Feng, Y., Hu, L., Su, H.: The Variance and Covariance of Fuzzy Random Variables. *Fuzzy Sets and Systems*, pp. 487–497 (2001)
3. Fortemps, P., Roubens, M.: Ranking and Defuzzification Methods Based on Area Compensation. *Fuzzy Sets and Systems*, pp. 319–330 (1996)
4. Kruse, R., Meyer, K.D.: *Statistics with Vague Data*. Riedel Publ. Co., Dordrecht (1987)
5. Inuiguchi, M., Tanino, T.: Portfolio Selection under Independent Possibilistic Information. *Fuzzy Sets and Systems*, pp. 83–92 (2000)
6. Klir, G.J., Yuan, B.: *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice-Hall, London (1995)
7. Kwakernaak, H.: Fuzzy Random Variables – I. Definitions and Theorem. *Inform. Sci.*, 1–29 (1978)
8. Markowitz, H.: *Mean-Variance Analysis in Portfolio Choice and Capital Markets*. Blackwell Publ., Oxford (1990)
9. Pliska, S.R.: *Introduction to Mathematical Finance: Discrete-Time Models*. Blackwell Publ., New York (1997)
10. Puri, M.L., Ralescu, D.A.: Fuzzy Random Variables. *J. Math. Anal. Appl.* 114, 409–422 (1986)
11. Yoshida, Y.: A Mean Estimation of Fuzzy Numbers by Evaluation Measures. In: Negoita, M.G., Howlett, R.J., Jain, L.C. (eds.) *KES 2004. LNCS (LNAI)*, vol. 3214, pp. 1222–1229. Springer, Heidelberg (2004)

12. Yoshida, Y.: Mean values, Measurement of Fuzziness and Variance of Fuzzy Random Variables for Fuzzy Optimization. In: Proc. SCIS & ISIS, pp. 2277–2282 (2006)
13. Yoshida, Y.: Fuzzy Extension of Estimations with Randomness: The Perception-based Approach. In: Torra, V., Narukawa, Y., Yoshida, Y. (eds.) MDAI 2007. LNCS (LNAI), vol. 4617, pp. 295–306. Springer, Heidelberg (2007)
14. Zadeh, L.A.: Fuzzy Sets. *Inform. and Control*, 338–353 (1965)
15. Zadeh, L.A.: Toward a Perception-Based Theory of Probabilistic Reasoning with Imprecise Probabilities. *J. Stat. Plan. Infer.*, 233–264 (2002)

Attribute Reduction Based on the Fuzzy Information Filter Operators

Fachao Li^{1,2}, Chao Gao², and Chenxia Jin¹

¹ School of Economics and Management, Hebei University of Science and Technology,
Shijiazhuang, Hebei, 050018, China

² School of Science, Hebei University of Science and Technology,
Shijiazhuang, Hebei, 050018, China

lifachao@tsinghua.org.cn, gaochao-19830202@163.com,
jinchenxia2005@126.com

Abstract. Fuzzy set theory and rough set theory are effective tools for dealing with incomplete and inaccurate knowledge of information systems. In this paper, for the difference of fuzzy equivalence relation matrix of the attributes in information system, by analyzing the shortcomings of the existing methods, we propose the axiomatic system of fuzzy information filter operators, and give several operational filter operators models, spell out an attribute reduction method based on fuzzy information filter operators (denoted by FIFO-RED for short); Finally, we analyze the characteristics and performance of FIFO-RED by a concrete example. And the results indicate that FIFO-RED can more effectively merge decision consciousness into information processing than FIE-RED, it has strong application value.

Keywords: Fuzzy equivalence matrix; Fuzzy information entropy; FIE-RED; Fuzzy information filter operator; FIFO-RED.

1 Introduction

In the real world, uncertainty is a widespread phenomenon and is unavoidable in many practical fields. How to deal with uncertain information by reasonable methods is the present focus in the field of artificial intelligence. For many years, researchers have been striving to find an effective ways to deal with uncertain information. In 1965, Zadeh established the fuzzy set theory; in the 1980s, Pawlak established the rough set theory, these theories not only laid a foundation for dealing with the various fuzzy information and incomplete information, but also have been widely used in machine learning, data mining, artificial intelligence, expert systems, pattern recognition, inductive inference, and so on.

However, Pawlak rough set model is only suitable for dealing with the nominal variables, and it can't directly deal with certain numerical data such as the spectrum signal [2] widely existing in vibration analysis, the temperature, current, voltage signal of condition analysis of transformers [3-4], the current signal of distribution system [5] and so on. For this problem, many researchers have made many useful explorations, here are some interesting literatures, [6] changed the numerical attributes into symbol

attributes by discretization method, but this method would unavoidably lead to the loss of information [7], and the results depend largely on the discretization. In order to solve this problem, literatures [8-12] introduced a fuzzy-rough set model and rough-fuzzy set model, further, they established the attributes reduction methods based on fuzzy information entropy (FIE-RED).

Although these methods have been successful in some specific areas, there are still many deficiencies in theory, for instance: ① When the discrimination of the data is not clear, we can not realize the reduction of information system; ② It is difficult to systematically merge uncertain decision consciousness into the process of reduction, so how to remove noise, that is, weaken the uncertainty of fuzzy information, is the key to make up these deficiencies.

Based on the above analysis, in this paper, we propose the axiomatic system of filter operators of fuzzy information, give several operational filter operators model with better structure, and establish an attribute reduction method based on information filter operators (FIFO-RED); Finally, we compare the performance of FIE-RED. And the results indicate that FIFO-RED can more effectively merge decision consciousness into information processing than FIE-RED, and it has a good structure characteristic and makes up the deficiencies of the fuzzy rough set model to some degree.

2 Preliminaries

2.1 Fuzzy Equivalence Class

Definition 1. Let R be a fuzzy relation over U , R is called a fuzzy equivalence relation, if R satisfies:

- 1) *Reflectivity*: $R(x, x) = 1$ for any $x \in U$;
- 2) *Symmetry*: $R(x, y) = R(y, x)$ for any $x, y \in X$;
- 3) *Transitivity*: $R(x, z) \geq \min\{R(x, y), R(y, z)\}$.

Obviously, when R_1, R_2 are fuzzy equivalence relations over U , $R_1 \cup R_2, R_1 \cap R_2$ are fuzzy equivalence relations, and $(R_1 \cup R_2)(x, y) = \max\{R_1(x, y), R_2(x, y)\}$, $(R_1 \cap R_2)(x, y) = \min\{R_1(x, y), R_2(x, y)\}$.

Definition 2. Let U be non-empty universe, and R a fuzzy equivalence relation on U . For given $x \in U$, the fuzzy set $R|_x$ is called a fuzzy equivalence class containing x , (denoted by $[x]_R$ for short), and $U/R = \{[x]_R \mid x \in U\}$ is called a fuzzy equivalence partition of U induced by R . Here, $R|_x(y) = R(x, y)$.

For given $U = \{x_1, x_2, \dots, x_n\}$ and the fuzzy equivalence relation R on U , if the relation matrix of R denoted by $M(R)$:

$$M(R) = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ r_{n1} & r_{n2} & \cdots & r_{nn} \end{pmatrix},$$

then for any $x_i \in U$, $[x_i]_R = r_{i1}/x_1 + r_{i2}/x_2 + \dots + r_{in}/x_n$. Where, $r_{ij} = R(x_i, x_j)$ is the relation value of x_i and x_j .

Theorem 1. Let R be a fuzzy equivalence relation on U , $x, y \in U$. Then

$$1) R(x, y) = 0 \Leftrightarrow [x]_R \cap [y]_R = \emptyset ; 2) R(x, y) = 1 \Leftrightarrow [x]_R = [y]_R .$$

Definition 3. Let $\langle U, A, V, f \rangle$ be a information system, $a \in B \subset A$. 1) If $U/B = U/(B-a)$, then a is redundant, otherwise, a is indispensable. 2) If the elements in B are all indispensable, then we say B is independent. 3) If B is independent and $U/B = U/A$, then we say B is a reduction of A .

Attributes reduction is the core content of information system, in the following, we will introduce the concept of fuzzy information entropy for reduction.

2.2 Fuzzy Information Entropy

Definition 4. Let $U = \{x_1, x_2, \dots, x_n\}$ be a non-empty universe, R be a fuzzy equivalence relation on U , $[x_i]_R$ be the fuzzy equivalence class containing x_i generated by R , then $|[x_i]_R| = \sum_{j=1}^n R(x_i, x_j)$ is called the cardinality of $[x_i]_R$;

$$H(R) = -\frac{1}{n} \sum_{i=1}^n \log_2 \frac{|[x_i]_R|}{n} \quad (1)$$

is called the information entropy of R .

Definition 5. Let $\langle U, A, V, f \rangle$ be a fuzzy information system, $U = \{x_1, x_2, \dots, x_n\}$, $B, E \subseteq A$, $[x_i]_B$ and $[x_i]_E$ be the fuzzy equivalence classes containing x_i generated by B and E . Then

$$H(BE) = -\frac{1}{n} \sum_{i=1}^n \log_2 \frac{|[x_i]_B \cap [x_i]_E|}{n} \quad (2)$$

is called hybrid entropy of B and E .

Considering we will use the value of the hybrid entropy in the process of attribute reduction, we will introduce the following theorems.

Theorem 2. Let $\langle U, A, V, f \rangle$ be a fuzzy information system, $A_i \in A, B \subset A$. 1) If A_i is redundant, then $H(A) = H(A - A_i)$; 2) If A is independent, then $H(A) > H(A - A_i)$; 3) If B satisfies $H(B) = H(A)$ and $H(B) > H(B - A_i)$ for any $A_i \in B$, then B is a reduction of A .

3 The Axiomatic System of Fuzzy Information Filter Operators

Although the membership function of the fuzzy set can describe the membership state between the element and set, it is difficult to completely consider it in reality. When

making decision, we often need to strengthen or weaken the influence of different components in uncertain information according to certain decision consciousness, this process is called information filtering. The information filtering is essentially a transformation from a set to another set, the guiding principal is that: in the condition of maintaining the basic relation characteristics of the sets, by adjusting the membership state between elements and set according to some principle, we weaken the uncertainty, and these will establish a basis for the decision-making.

According to the basic characteristics of fuzzy decision-making (It is hardest to make decision when membership degree is 0.5) as well as the above analysis, the fuzzy information filtering operator (the mapping from $\mathcal{F}(X)$ to $\mathcal{F}(X)$) should meet the following basic principles:

Principle 1 (Normality): If $A(x)=0$, then $F(A)(x)=0$; if $A(x)=0.5$, then $F(A)(x)=0.5$; if $A(x)=1$, then $F(A)(x)=1$;

Principle 2 (Monotonicity): If $A(x_1) \leq A(x_2)$, then $F(A)(x_1) \leq F(A)(x_2)$;

Principle 3 (Filterability): If $0 < A(x) < 0.5$, then $F(A)(x) \leq A(x)$; if $0.5 < A(x) < 1$, then $F(A)(x) \geq A(x)$.

Principle 1 shows that some typical characteristics of fuzzy information will not change after information filtering. Principle 2 shows filtering process do not change the basic relation feature between elements and set. Principle 3 shows that the information filtering process weakened the fuzziness of information. For simplicity, we call the mapping F from $\mathcal{F}(X)$ to $\mathcal{F}(X)$ fuzzy information filter operator.

It is easily prove that, for any $0 < \delta < 0.5$:

$$F_1(A)(x) = \begin{cases} 0 & A(x) < \delta \\ A(x) & A(x) \geq \delta \end{cases} \tag{3}$$

$$F_2(A)(x) = \begin{cases} A(x) & A(x) \leq 1 - \delta \\ 1 & A(x) > 1 - \delta \end{cases} \tag{4}$$

$$F_3(A)(x) = \begin{cases} 0 & A(x) < \delta \\ A(x) & \delta \leq A(x) \leq 1 - \delta \\ 1 & A(x) > 1 - \delta \end{cases} \tag{5}$$

$$F_4(A)(x) = \begin{cases} 0 & 0 \leq A(x) < \delta \\ h(A(x)) & \delta \leq A(x) \leq 1 - \delta \\ 1 & A(x) > 1 - \delta \end{cases} \tag{6}$$

are all fuzzy information filter operators (where $h(t) = (1 - 2\delta)^{-1}(t - \delta)$), and we call (3) δ -below filtering operator, (4) δ -upper filtering operator, (5) δ -filtering operator, (6) δ -linear filtering operator.

Theorem 3. $G(X)$ is a function from $[0,1]$ to $[0,1]$, if it satisfies: 1) Monotone non-decreasing; 2) $G(0) = 0$, $G(0.5) = 0.5$, $G(1) = 1$; 3) $G(x) \leq x$ for each $0 < x < 0.5$, $G(x) \geq x$ for each $0.5 < x < 1$, then $G(A)$ (that is, $G(A)(x) = G(A(x))$) is fuzzy information filter operator induced by $G(x)$.

For any given real numbers a, b, c, d and $0 \leq b \leq a \leq 0.5 \leq c \leq d \leq 1$, we can prove that $G(x)$ (shown as Fig. 1):

$$G(x) = \begin{cases} ba^{\frac{1}{a}}x & 0 \leq x < a \\ 0.5 - (0.5 - b)(0.5 - a)^{-1}(0.5 - x) & a \leq x < 0.5 \\ 0.5 + (d - 0.5)(c - 0.5)^{-1}(x - 0.5) & 0.5 \leq x < c \\ 1 - (1 - d)(1 - c)^{-1}(1 - x) & c \leq x \leq 1 \end{cases} \quad (7)$$

satisfies the conditions in Theorem 4. $G(x)$ is constituted by four liner functions, so we call $G(A)$ induced by (7) quasi-linear fuzzy information filtering operators. In particular, when $0.5 - a = c - 0.5$ (that is, $a + c = 1$) and $b = 1 - d$ (that is, $b + d = 1$), we call $G(A)$ induced by (7) symmetric quasi-linear fuzzy information filtering operator.

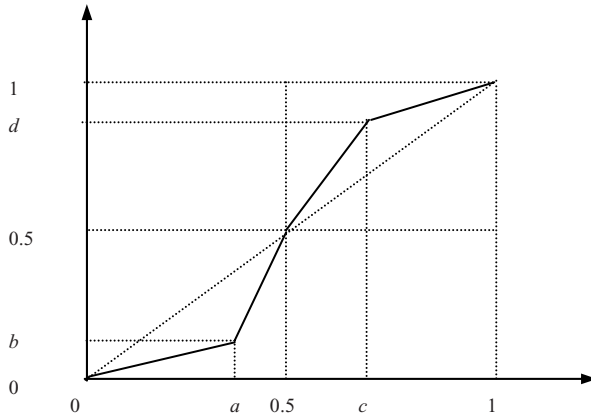


Fig. 1. The sketch map of quasi-linear fuzzy information filtering operator

By using fuzzy metric theory, we know that, for any filtering operator F and any $A \in \mathcal{F}(X)$, the fuzziness of $F(A)$ does not exceed that of A , it shows the above fuzzy information filtering methods have the role of weakening the uncertainty.

It is easy to see that: ① In (3) ~ (6), δ is a parameter which reflects the filtering strength, the larger δ is, the larger the filter strength is; ② In (7), a, b, c, d are parameters reflecting the filtering strength, the more closer to 0.5 the value of a and c are, the more closer to 0 the value of b is, and the more closer to 1 the value of d is, the larger the filtering strength is; the more closer to 0.5 the value of a, b are, and the more closer to 1 the value of c, d are, the smaller the filtering strength is.

But it is worth noting that, larger filtering strength does not mean better filtering results, therefore, we should select appropriate fuzzy information filtering operators by combining decision view.

4 Attribute Reduction Based on the Information Filter Operators

Attribute reduction is a main application of rough set theory in pattern recognition, data mining etc.. As we know, finding the minimal reduction of a knowledge base is a NP hard problem, and the existing heuristic reduction algorithm based on significant measure of the attributes mainly construct the reduction containing the minimal condition attributes, these algorithms get a suboptimal result but relatively low time-consuming. In the following we will introduce a new kind of attribute reduction algorithm based on the fuzzy information filter operators, we firstly give significance of attribute.

Definition 7. Let $\langle U, A, V, f \rangle$ be a fuzzy information system, $A_i \in B \subset A$, the significance of attribute A_i in attribute set B is defined as $\text{sig}(A_i, B) = H(B) - H(B - A_i)$.

Base on the above measures, a greedy algorithm based on the fuzzy information filter operator for computing reduction can be constructed (denoted by FIFO-RED for short). In the following, we will give the concrete implementation steps:

Step 1. Input information system $IS = \langle U, A, V, f \rangle$.

Step 2. $\forall A_i \in A$: compute the fuzzy equivalence matrix, choose appropriate fuzzy information filter operator F to filter the fuzzy equivalence matrix;

Step 3. $red = \emptyset$;

Step 4. For each $A_i \in A - red$, compute significance of attribute A_i in attribute set $A_i \cup red$, $\text{sig}(a_i, a_i \cup red) = H(a_i \cup red) - H(a_i)$;

Step 5. Choose attribute A_K which satisfies: $G(A_K) = \max_i [SIG(A_i, A_i \cup red)]$;

Step 6. If $G(A_K) > 0$, then $red = red \cup \{A_K\}$ goto step 4, else go to step 7;

Step 7. Output the reduct of IS.

5 Experiments and Analysis

Example. Let $X = \{x_1, x_2, x_3, x_4, x_5, x_6\}$, $A = \{A_1, A_2, A_3, \dots, A_{10}\}$, M_i is the fuzzy relation determined by A_i , $i = 1, 2, \dots, 10$. Try to determine the reduction of the attribute set A . Here,

$$\begin{aligned}
 M_1 &= \begin{pmatrix} 1 & 0.1 & 0.2 & 0.3 & 0.7 & 0.2 \\ 0.1 & 1 & 0.3 & 0.8 & 0.6 & 0.2 \\ 0.2 & 0.3 & 1 & 0.3 & 0.5 & 0.6 \\ 0.3 & 0.8 & 0.3 & 1 & 0.2 & 0.2 \\ 0.7 & 0.6 & 0.5 & 0.2 & 1 & 0.3 \\ 0.2 & 0.2 & 0.6 & 0.2 & 0.3 & 1 \end{pmatrix}, M_2 = \begin{pmatrix} 1 & 0.3 & 0.1 & 0.3 & 0.7 & 0.2 \\ 0.3 & 1 & 0.3 & 0.8 & 0.6 & 0.2 \\ 0.1 & 0.3 & 1 & 0.3 & 0.5 & 0.6 \\ 0.3 & 0.8 & 0.3 & 1 & 0.2 & 0.2 \\ 0.7 & 0.6 & 0.5 & 0.2 & 1 & 0.3 \\ 0.2 & 0.2 & 0.6 & 0.2 & 0.3 & 1 \end{pmatrix}, M_3 = \begin{pmatrix} 1 & 0.2 & 0.2 & 0.1 & 0.7 & 0.2 \\ 0.2 & 1 & 0.3 & 0.8 & 0.6 & 0.2 \\ 0.2 & 0.3 & 1 & 0.3 & 0.5 & 0.6 \\ 0.1 & 0.8 & 0.3 & 1 & 0.2 & 0.2 \\ 0.7 & 0.6 & 0.5 & 0.2 & 1 & 0.3 \\ 0.2 & 0.2 & 0.6 & 0.2 & 0.3 & 1 \end{pmatrix}, M_4 = \begin{pmatrix} 1 & 0.3 & 0.2 & 0.3 & 0.65 & 0.2 \\ 0.3 & 1 & 0.3 & 0.8 & 0.6 & 0.2 \\ 0.2 & 0.3 & 1 & 0.3 & 0.5 & 0.6 \\ 0.3 & 0.8 & 0.3 & 1 & 0.2 & 0.2 \\ 0.65 & 0.6 & 0.5 & 0.2 & 1 & 0.3 \\ 0.2 & 0.2 & 0.6 & 0.2 & 0.3 & 1 \end{pmatrix}, \\
 M_5 &= \begin{pmatrix} 1 & 0.2 & 0.2 & 0.3 & 0.7 & 0.1 \\ 0.2 & 1 & 0.3 & 0.8 & 0.6 & 0.2 \\ 0.2 & 0.3 & 1 & 0.3 & 0.5 & 0.6 \\ 0.3 & 0.8 & 0.3 & 1 & 0.2 & 0.2 \\ 0.7 & 0.6 & 0.5 & 0.2 & 1 & 0.3 \\ 0.1 & 0.2 & 0.6 & 0.2 & 0.3 & 1 \end{pmatrix}, M_6 = \begin{pmatrix} 1 & 0.2 & 0.2 & 0.3 & 0.7 & 0.2 \\ 0.2 & 1 & 0.1 & 0.8 & 0.6 & 0.2 \\ 0.2 & 0.1 & 1 & 0.3 & 0.5 & 0.6 \\ 0.3 & 0.8 & 0.3 & 1 & 0.2 & 0.2 \\ 0.7 & 0.6 & 0.5 & 0.2 & 1 & 0.3 \\ 0.2 & 0.2 & 0.6 & 0.2 & 0.3 & 1 \end{pmatrix}, M_7 = \begin{pmatrix} 1 & 0.2 & 0.2 & 0.3 & 0.7 & 0.2 \\ 0.2 & 1 & 0.3 & 0.75 & 0.6 & 0.2 \\ 0.2 & 0.3 & 1 & 0.3 & 0.5 & 0.6 \\ 0.3 & 0.75 & 0.3 & 1 & 0.2 & 0.2 \\ 0.7 & 0.6 & 0.5 & 0.2 & 1 & 0.3 \\ 0.2 & 0.2 & 0.6 & 0.2 & 0.3 & 1 \end{pmatrix}, M_8 = \begin{pmatrix} 1 & 0.3 & 0.2 & 0.3 & 0.7 & 0.2 \\ 0.3 & 1 & 0.3 & 0.8 & 0.55 & 0.2 \\ 0.2 & 0.3 & 1 & 0.3 & 0.5 & 0.6 \\ 0.3 & 0.8 & 0.3 & 1 & 0.2 & 0.2 \\ 0.7 & 0.55 & 0.5 & 0.2 & 1 & 0.3 \\ 0.2 & 0.2 & 0.6 & 0.2 & 0.3 & 1 \end{pmatrix},
 \end{aligned}$$

$$M_9 = \begin{pmatrix} 1 & 0.2 & 0.2 & 0.3 & 0.7 & 0.2 \\ 0.2 & 1 & 0.3 & 0.8 & 0.6 & 0.2 \\ 0.2 & 0.3 & 1 & 0.3 & 0.5 & 0.6 \\ 0.3 & 0.8 & 0.3 & 1 & 0.2 & 0.2 \\ 0.7 & 0.6 & 0.5 & 0.2 & 1 & 0.3 \\ 0.2 & 0.2 & 0.6 & 0.2 & 0.3 & 1 \end{pmatrix}, M_{10} = \begin{pmatrix} 1 & 0.6 & 0.2 & 0.3 & 0.7 & 0.2 \\ 0.6 & 1 & 0.3 & 0.8 & 0.6 & 0.2 \\ 0.2 & 0.3 & 1 & 0.3 & 0.3 & 0.6 \\ 0.3 & 0.8 & 0.3 & 1 & 0.2 & 0.2 \\ 0.7 & 0.6 & 0.3 & 0.2 & 1 & 0.3 \\ 0.2 & 0.2 & 0.6 & 0.2 & 0.3 & 1 \end{pmatrix}.$$

We separately use the reduction algorithm (FIE-RED) of the literature [11] and FIFO-RED to analyze the above example:

Case 1. By using FIE-RED, the result is $red = \{A_i | i = 1, 2, \dots, 10\}$, that is, these matrixes can not be reduced by FIE-RED, because the algorithm admit a hybrid information entropy, and each one of the above mentioned 10 matrixes has a minimal element in different location. The hybrid information entropy is only accordance with counterparts elements for small, each of the above are useful matrix, so we can not reduction, the result is also not what we want to see.

Case 2. The result of FIFO-RED is shown in Table 1.

Table 1. The results of FIFO-RED

filter operators	parameter	reduction
(3)	$\delta = 0.2$	$\{A_3, A_{10}, A_8, A_4, A_6, A_1, A_5, A_2, A_9\}$
	$\delta = 0.3$	$\{A_3, A_8, A_6, A_{10}, A_4, A_7\}$
	$\delta = 0.4$	$\{A_8, A_{10}, A_4, A_7\}$
(4)	$\delta = 0.2$	$\{A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, A_{10}\}$
	$\delta = 0.3$	$\{A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, A_{10}\}$
	$\delta = 0.4$	$\{A_3, A_6, A_{10}, A_2, A_5, A_9, A_4, A_{10}\}$
(5)	$\delta = 0.2$	$\{A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, A_{10}\}$
	$\delta = 0.3$	$\{A_3, A_8, A_6, A_{10}, A_4, \}$
	$\delta = 0.4$	$\{A_8, A_{10}\}$
(6)	$\delta = 0.2$	$\{A_3, A_8, A_{10}, A_4, A_7, A_6\}$
	$\delta = 0.3$	$\{A_8, A_{10}, A_4\}$
	$\delta = 0.4$	$\{A_8, A_{10}\}$
(7)	$a = 0.49, b = 0.1$ $c = 0.51, d = 0.99$	$\{A_8, A_{10}, A_3, A_4, A_7, A_6, A_9, A_2\}$
	$a = 0.49, b = 0.45$ $c = 0.88, d = 0.99$	$\{A_3, A_8, A_{10}, A_6, A_4, A_2, A_5\}$
	$a = 0.3, b = 0.1$ $c = 0.8, d = 0.9$	$\{A_8, A_{10}, A_3, A_4, A_7, \}$

We can see from the reduction results by filter operators (3) ~ (7) in Table1 that:

- ① In the same filter model, the larger the filter strength is, the less the reduction results is;
- ② The reduction results will vary with different filter operators;
- ③ Despite

the change of the filter operators and filter strength, some attributes, for instance, A_8 and A_{10} always exist in reduction results of FIFO-RED.

6 Conclusion

We propose the axiomatic system of fuzzy information filter operators in this paper, spell out an FIFO-RED algorithm, compared the performance of FIE-RED by a concrete example. The results indicate, 1) The axiomatic system has given quantitative and qualitative information filtering principle. 2) FIFO-RED can merge decision consciousness into information processing according to certain principle, and it can also effectively realize the reduction of information system without losing the core attributes, we can see that FIFO-RED is more effective than FIE-RED, and FIFO-RED can effectively make up the deficiencies of the fuzzy rough set model.

Acknowledgements. This work is supported by the National Natural Science Foundation of China (70671034) and the Ph. D. Foundation of Hebei Province (05547004D-2, B2004509) and the Natural Science Foundation of Hebei Province (F2006000346).

References

1. Radzikowska, A.M., Kerre, E.E.: A Comparative Study of Fuzzy Rough Set. *Fuzzy Sets and Systems*, 137–156 (2002)
2. Yu, D.R., Hu, Q.H., Bao, W.: Combining Rough Set Methodology and Fuzzy Clustering for Knowledge Discovery from Quantitative Data. *Proceedings of the Chinese Society for Electrical Engineering*, 205–210 (2004)
3. Zhu, Y.L., Wu, L.Z., Li, X.Y.: Synthesized Diagnosis on Transformer Faults Based on Bayesian Classifier and Rough Set. *Proceedings of the Chinese Society for Electrical Engineering*, 159–165 (2005)
4. Wang, Y.Q., Li, F.C., Li, H.M.: Synthetic Fault Diagnosis Method of Power Transformer Based on Rough Set Theory and Bayesian Network. *Proceedings of the Chinese Society for Electrical Engineering*, 137–141 (2006)
5. Sun, Q.Y., Zhang, H.G.: Fault Diagnose Algorithm of Distribution System by Continuous Signals Based on Rough Sets. *Proceedings of the Chinese Society for Electrical Engineering*, 156–161 (2006)
6. Xie, H., Cheng, H.Z., Niu, D.X.: Discretization of Continuous Attributes in Rough Set Theory Based on Information Entropy. *Chinese Journal of Computers*, 1570–1574 (2005)
7. Jensen, R., Shen, Q.: Semantics-preserving Dimensionality Reduction: Rough and Fuzzy-rough Fuzzy-rough-based Approaches. *IEEE Transactions on Knowledge and Data Engineering*, 1457–1471 (2004)
8. Dubois, D., Prade, H.: Rough Fuzzy Sets and Fuzzy Rough Sets. *International Journal General Systems*, 191–209 (1990)
9. Hu, Q.H., Yu, D.R., Xie, Z.X., Liu, J.F.: Fuzzy Probabilistic Approximation Spaces and Their Information Measures. *IEEE transactions on Fuzzy Systems*, 191–201 (2006)

10. Yeung, D.S., Chen, D.G., Tsang, E.C.C., Lee, J.W.T., Wang, X.Z.: On the Generalization of Fuzzy Rough Sets. *IEEE Transactions on Fuzzy Systems*, 343–361 (2005)
11. Hu, Q.H., Yu, D.R., Xie, Z.X.: Information-preserving Hybrid Data Reduction Based on Fuzzy-rough Techniques. *Pattern Recognition Letters*, 414–423 (2006)
12. Slowinski, R., Vanderpooten, D.: A Generalized Definition of Rough Approximations Based on Similarity. *IEEE Transactions on Knowledge and Data Engineering*, 331–336 (2000)

Classification of Image Data Using Gradient-Based Fuzzy C-Means with Mercer Kernel

Dong-Chul Park

Center for Intelligent Imaging Systems Research,
Myong Ji University, Korea
parkdc@mju.ac.kr

Abstract. A novel approach for the classification of image signals for image retrieval using Gradient-Based Fuzzy C-Means with Mercer Kernel (GBFCM-MK) is proposed and presented in this paper. The proposed classifier is a FCM-based algorithm which utilizes the Mercer Kernel to exploit the statistical nature of the image data to improve the classification accuracy. Experiments and results on various data sets demonstrate that the proposed classification algorithm outperforms 21.7% - 24% in accuracy in comparison with conventional algorithms such as the traditional Fuzzy C-Means (FCM), Gradient-based Fuzzy C-Means (GBFCM), and GBFCM with Divergence Measure (GBFCM(DM)).

1 Introduction

With the large number of digital image data are stored in databases as compressed formats such as jpeg, png, and bmp, automatic classification and retrieval methods have received much attention recently from researchers in this field. The rapid increase in the amount of image data demands for a computerized method which allows efficient and automated content-based classification and retrieval of image data. In a content-based retrieval system for image data, image classification normally involves the processing of two main tasks: feature extraction and a classification task that uses the extracted features to discriminate the classes. When a perfect set of features that can describe the image data is given, the accuracy of the resultant classification depends on the classifier adopted.

Many clustering algorithms have been successfully applied in this field. Traditionally, conventional clustering algorithms such as the Self Organizing Map (SOM)[\[1\]](#) and the K-Means[\[2\]](#) have been most widely used in practice. But they assign an object to a single class and ignore the possibility that the object may also belong to other classes. Fuzzy clustering techniques have been proposed for clustering problems. One of the most widely used algorithms employing fuzzy clustering techniques is the Fuzzy C-Means (FCM) algorithms.

Gradient Based Fuzzy C-Means (GBFCM) algorithm is solving the FCM disadvantage that each iteration requires the use of all the data at once. GBFCM

is introduced by Park[3]. GBFCM combines the characteristics of Kohonen network (presenting one data at a time and applying the gradient descent method) and the FCM algorithm (continuous values of the membership grades in the range [0, 1]). In the FCM algorithm, all the data are present in the objective function, and the gradients are set to zero in order to obtain the equations necessary for minimization[4]. While only one datum at a time is required in the GBFCM. Recently, the GBFCM is expanded to a version with Divergence Measure(DM) to deal with Gaussian Probability Distribution Function (GPDF) data [5]. The GBFCM(DM) exploits advantages of clustering GPDF data and shows an improvement in classification accuracy over SOM and FCM.

In this paper, we propose a novel classification model which can better exploit the statistical nature of image data by employing the GBFCM algorithm with Mercer Kernel (GBFCM-MK). While the GBFCM algorithm has been proven to give high accuracy of clustering[3], the Mercer Kernel was also proven to give better modeling of statistic data like image data. Therefore, this combination implies an improvement for image classification in terms of classification accuracy.

The remainder of this paper is organized as follows: Section 2 briefly summarizes the FCM and GBFCM and the Gradient Based Fuzzy C-Means algorithm with Mercer Kernel(GBFCM-MK) is outlined in Section 4. Section 5 provides the experiments involving practical image data set and presents the results. Finally, conclusions are given in Section 7.

2 Gradient-Based Fuzzy c-Means(GBFCM) Algorithm

2.1 Fuzzy c-Means(FCM) Algorithm

For FCM, the objective function is defined as :

$$J_m(U, \mathbf{v}) = \sum_{k=1}^n \sum_{i=1}^M (\mu_{ki})^m (d_i(\mathbf{x}_k))^2 \tag{1}$$

where $d_i(\mathbf{x}_k)$ denotes the distance from the input data \mathbf{x}_k to \mathbf{v}_i , the center of the cluster i , μ_{ki} is the membership value of the data \mathbf{x}_k to the cluster i , and m is the weighting exponent, $m \in 1, \dots, \infty$, while n and M are the number of input data and clusters, respectively. Note that the distance measure used in FCM is the Euclidean distance.

Bezdek defined a condition for minimizing the objective function with the following two equations [6]:

$$\mu_{ki} = \frac{1}{\sum_{j=1}^M \left(\frac{d_i(\mathbf{x}_k)}{d_j(\mathbf{x}_k)}\right)^{\frac{2}{m-1}}} \tag{2}$$

$$\mathbf{v}_i = \frac{\sum_{k=1}^n (\mu_{ki})^m \mathbf{x}_k}{\sum_{k=1}^n (\mu_{ki})^m} \tag{3}$$

The FCM finds the optimal values of group centers iteratively by applying Eq. (2) and Eq. (3) in an alternating fashion.

2.2 Gradient-Based Fuzzy C-Means(GBFCM) Algorithm

One attempt to improve the FCM algorithm was made by minimizing the objective function using one input data at a time instead of the entire training data. That is, the FCM in Eq. (2) and Eq. (3) uses all data to update the center value of the cluster, but the GBFCM that is used in this paper was developed to update the center value of the cluster with a given individual data sequentially [37].

The update equations for \mathbf{v}_i and μ_{ki} in GBFCM are as following:

$$\mathbf{v}_i(n + 1) = \mathbf{v}_i(n) - 2\eta\mu_{ki}^2(\mathbf{v}_i(n) - \mathbf{x}_k), \quad i = 1, 2, \dots, M \tag{4}$$

$$\mu_{ki} = \frac{1}{\sum_{j=1}^M \left(\frac{d_i(\mathbf{x}_k)}{d_j(\mathbf{x}_k)}\right)^2} \tag{5}$$

More detailed explanation about the GBFCM can be found in [37].

3 Gradient Based Fuzzy C-Means with a Mercer Kernel

In the case of the Gaussian kernel function, the objective function can be rewritten as:

$$J_i^\Phi = 2 \sum_{k=1}^M \mu_{ki}^2 \left(1 - e^{-\frac{\|\mathbf{v}_k - \mathbf{x}_i\|^2}{\sigma^2}}\right) \tag{6}$$

By using the steepest gradient descent algorithm, the objective function can be minimized and we obtain:

$$\mathbf{v}_{k+1} = \mathbf{v}_k - \eta\mu_{ki}^2\sigma^{-2}K(\mathbf{v}_k, \mathbf{x}_i)(\mathbf{v}_k - \mathbf{x}_i) \tag{7}$$

where η is a learning gain.

The constrained optimization in our problem can be solved by using the Lagrange multiplier.

$$F_m = 2 \sum_{k=1}^M \mu_{ki}^2 (1 - K(\mathbf{v}_k, \mathbf{x}_i)) - \lambda \left(\sum_{k=1}^M \mu_{ki}^2 - 1\right) \tag{8}$$

By taking the first derivative of F_m with respect to μ_{ki} , setting the result to zero, and using the constraint $\sum_{j=1}^M \mu_{ji} = 1$, the membership grades can be obtained as follows [8]:

$$\mu_{ki} = \frac{1}{\sum_{j=1}^M \left(\frac{1 - K(\mathbf{v}_k, \mathbf{x}_i)}{1 - K(\mathbf{v}_j, \mathbf{x}_i)}\right)} \tag{9}$$

More detailed explanation about the GBFCM-MK can be found in [8].

4 Experiments and Results

For experiments, image data sets are collected from the Caltech image data set. The Caltech image data set consists of different image classes (categories) in which each class contains different views of an object. The Caltech image data were collected by the Computational Vision Group and are available at the following website :

<http://www.vision.caltech.edu/html-files/archive.html>

Figs. 1(a), 1(b), 1(c), 1(d), and 1(e) show examples of airplane, car, motorbike, bike, and face used in our experiments, respectively. Each class consists of 150 images with different views resulting in a total of 750 images in the data set. From this data set, 100 images were randomly chosen for training classifiers while the remaining images were used for testing classifiers. Before any further



(a)



(b)



(c)



(d)



(e)

Fig. 1. (a) Face (b) Airplane (c) Car (d) Motorbike (e) Bike

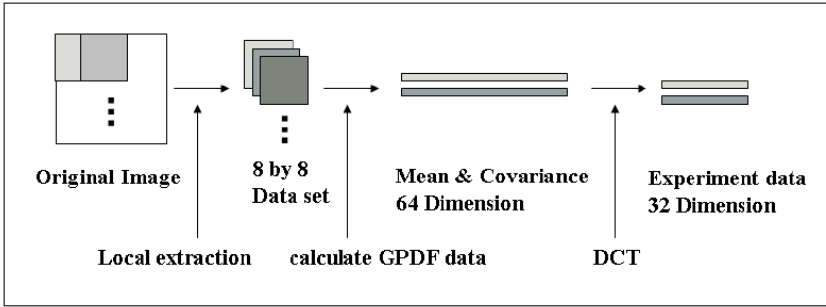


Fig. 2. Data processing procedure

processing for feature extraction, the entire image sets were converted to grey scale data with the same resolution.

In order to describe the texture information of images, the localized image representation method is employed. The localized representation method represents the content of the image by a collection of local features extracted from the image. These features are computed at different points of interest in the image. Afterwards, the Gaussian distribution, wherein the mean vector and the covariance matrix are estimated from all local feature vectors obtained from the image, is used to represent the content of the image. Localized representation maintains the dimensions of the feature vectors tractable by adjusting the sizes of blocks. This method is consequently more robust to occlusions and clutters.

In order to obtain the texture information from the image, conventional texture descriptors based on a frequency domain analysis such as Gabor filters [9] and wavelet filters [10] are often used. However, these algorithms often require a high computational load for feature extraction and are not suitable for real-time applications. In this paper, the Discrete Cosine Transform (DCT) is adopted for extracting the texture information from each block of the image [11]. The DCT transforms the image from the spatial domain into the frequency domain.

For the localized representation, images are transformed into a collection of 8×8 blocks. Each block is then shifted by an increment of 2 pixels horizontally and vertically. The DCT coefficients of each block are then computed and return in 64 dimensional coefficients. Only 32 lowest frequency DCT coefficients that are visible to the human eye are kept. Therefore, the feature vectors that are obtained from each block have 32 dimensions. In order to calculate the GPDF for the image, the mean vector and the covariance matrix are estimated from all blocks obtained from the image. Finally, a GPDF with a 32-dimensional mean vector and a 32×32 covariance matrix is used to represent the content of the image. Fig. 2 summarizes the procedure for the preparation used in our experiment [5].

For each image class, we model the distribution of its feature vectors by a number of code vectors. Each code vector represents a group with its own mean and covariance matrix. After that, during testing, the class of each image is

Table 1. The comparison of classification accuracy (%) of face/non-face classifier using different algorithm (FCM, GBFCM, GBFCM(DM), GBFCM-MK) and 4 code vectors

	face	non-face	Overall
FCM	91.62	56.28	73.95
GBFCM	90.21	78.69	84.45
GBFCM(DM)	94.91	79.25	87.08
GBFCM-MK	95.76	92.34	94.05

decided by using a classifier. The classifier decides the classes for each image by calculating the probability that an image belongs to every class, and then chose the cluster group which gives the shortest distance through the kernel.

$$Cluster\ group(x) = \arg\min_i (1 - Kernel(x, v_i)) \tag{10}$$

$$Kernel(x, v_i) = e^{-\frac{\|x - v_i\|^2}{\sigma^2}} \tag{11}$$

First, the face/non-face classification task was performed. In this case, 150 face data and 150 randomly selected non-face data from airplane, car, motorbike, and bike were employed. Each of the two data sets is partitioned into two subsets: 100 randomly selected samples are used for the training data set and the remaining 50 samples are used for the test data set. Experiments are performed with 50 different combinations of training and test data sets. The classification accuracies shown in Table 2 are the means of the obtained accuracies from the 50 different combinations of the training and test data sets. As can be seen from Table 2 for the face/non-face classification task, all four algorithms fairly distinguish face images from non-face images. However, only the GBFCM-MK successfully discriminates non-face images from face images. This implies that the Mercer kernel for GPDF data is a critical addition to GBFCM for the face/non-face classification task.

Another experiment related to the non-face data classification problem was performed. Each of 4 data sets, airplane, car, bike, motorbike, is again partitioned into two subsets: 100 randomly selected samples for the training data set and the remaining 50 samples for the test data set. Experiments are performed with 50 different combinations of training and test data sets. The classification accuracies reported in Table 3 are the means of the obtained accuracies from the 50 different cases of the training and test data sets. The average accuracy of FCM, GBFCM, GBFCM(DM), and GBFCM-MK over all numbers of code vectors are 67.1%, 69.4%, 74.1%, and 91.1%, respectively. Note that the FCM, GBFCM, and GBFCM(DM) algorithms don't use a kernel. From the result shown in Table 3, we can infer that the algorithm which uses the Kernel, GBFCM-MK, usually outperforms the FCM and GBFCM which use the Euclidean distance as its distance measure. Even with the GBFCM(DM) which uses a divergence as its distance measure, the accuracy is far behind the GBFCM-MK and the kernel method shows its significance in classifier performance. The results also show that GBFCM-MK gives

Table 2. Classification accuracy (%) of different algorithms

		Airplane	Car	Bike	Motorbike	Overall
FCM	mean	62.4	80.6	55.7	69.8	67.1%
	σ	21.9	23.6	20.7	14.8	$\sigma=20.3$
GBFCM	mean	70.4	67.6	66.3	73.3	69.4%
	σ	17.9	21.0	14.2	14.9	$\sigma=17.0$
GBFCM(DM)	mean	74.2	81.1	79.6	71.1	74.1%
	σ	8.6	17.2	12.7	13.6	$\sigma=13.2$
GBFCM-MK	mean	90.6	91.7	88.4	93.9	91.1%
	σ	9.8	14.1	11.5	8.8	$\sigma=11.1$

21.7% - 24.0% better accuracy over the other three algorithms. When compared to the face/non-face classification task, the this classification problem is considerably more complicated. However, the GBFCM-MK still outperforms the GBFCM by 21.7% in terms of classification accuracy and the usefulness of the Mercer kernel in the GBFCM is again demonstrated by this experiment.

5 Conclusion

In this paper, a new approach for the classification of image classification using a clustering algorithm is proposed. This paper demonstrates how image data can be classified using the Gradient-Based Fuzzy C-Means algorithm with Mercer Kernel (GBFCM-MK). Integration of the GBFCM and the Mercer kernel provides a better solution with non-linear vectors by mapping feature vectors into a high-dimensional feature space.

Experiments were performed for five data sets, a face data set and four non-face data set, consisting of 750 image data. Experiments on the face/non-face classification task were first performed. The results show that all four algorithms (FCM, GBFCM, GBFCM(MK), GBFCM-MK) fairly identify face data from non-face data. However, only the GBFCM-MK successfully discriminates non-face data from face data. In the subsequent experiments, another image classification problem was considered. Four non-face classes,(airplane, car, bike, motorbike), were trained and tested using FCM, GBFCM, GBFCM(MK), and GBFCM-MK. Results shows that the GBFCM-MK outperforms the FCM, GBFCM, and GBFCM(MK) by 24.0% and 21.7%, and 17.0%, respectively, in terms of average classification accuracy. The experiments and results clearly show that integration of the GBFCM and the Mercer kernel provides an efficient tool for the image data classification problem.

Acknowledgments. This work was supported by the Korea Science and Engineering Foundation (KOSEF) grant funded by the Korean government (MOST) (Grant No.: R01-2007-000-20330-0)

References

1. Kohonen, T.: The Self-Organizing Map. Proc. IEEE 78, 1464–1480 (1990)
2. Hartigan, J.: Clustering Algorithms. Wiley, New York (1975)
3. Park, D.C., Dagher, I.: Gradient Based Fuzzy c-means (GBFCM) Algorithm. In: IEEE Int. Conf. on Neural Networks, ICNN 1994, pp. 1626–1631 (1994)
4. Windham, M.P.: Cluster Validity for the Fuzzy c-means clustering algorithm. IEEE Trans. Pattern Anal. Mach. Int., 357–363 (1982)
5. Huang, V.T.L., Park, D.C., Woo, D.M.: Image Classification Using Gradient-based Fuzzy C-Means with Divergence Measure. In: IEEE Int. Joint Conf. on Neural Networks, Hong Kong (June 2008) (Accepted for presentation)
6. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York (1981)
7. Looney, C.: Pattern Recognition Using Neural Networks, pp. 252–254. Oxford University press, New York (1997)
8. Park, D.-C., Tran, C.N., Min, B.-J., Park, S.: Modeling and Classification of Audio Signals Using Gradient-based FCM Algorithm with a Mercer Kernel. In: Yang, Q., Webb, G. (eds.) PRICAI 2006. LNCS (LNAI), vol. 4099, pp. 1104–1108. Springer, Heidelberg (2006)
9. Daugman, J.G.: Complete Discrete 2D Gabor Transform by Neural Networks for Image Analysis and Compression. IEEE Trans. Acoust., Speech, Signal Process 36, 1169–1179 (1988)
10. Pun, C.M., Lee, M.C.: Extraction of Shift Invariant Wavelet Features for Classification of Images with Different Sizes. IEEE Trans. Pattern Anal. Mach. Intel. 26(9), 1228–1233 (2004)
11. Huang, Y.L., Chang, R.F.: Texture Features for DCT-Coded Image Retrieval and Classification. In: ICASSP Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, vol. 6, pp. 3013–3016 (1999)

Design of T-S Fuzzy Model Based on PSODE Algorithm

Ben Niu and Li Li

School of Management, Shenzhen University,
Shenzhen, 518060 P.R. China
drniuben@gmail.com, llii318@163.com

Abstract. This paper presents a novel fuzzy modeling strategy based on the hybrid of particle swarm optimization (PSO) and differential evolution (DE), and the proposed hybrid algorithm is referred to as PSODE. PSODE is based on a two-population scheme, in which the individuals in one population is enhanced by PSO and the individuals in the other population is evolved by DE. The individuals both in PSO and DE are co-evolved during the algorithm execution by employing an information sharing mechanisms. To further improve the proposed PSODE algorithm a nonlinear inertia weight approach and a mutation mechanism are presented respectively. In the simulation part, the PSODE is used to automatic design of fuzzy identifier for a nonlinear dynamic system. The performance of the suggested method is compared to PSO, DE and some other methods in the fuzzy identifier design to demonstrate its superiority.

Keywords: particle swarm; differential evolution; T-S; fuzzy model.

1 Introduction

Recently, Takagi-Sugeno (T-S) type fuzzy controllers have been successfully applied to the stabilization control design of nonlinear systems [1, 2]. The fuzzy systems were thought of as universal approximators for nonlinear dynamical systems, and the T-S fuzzy model has been proved to be a very good representation for a certain class of nonlinear dynamical systems. However, despite the previous successfully history, the lack of learning capabilities characterizing most of the works in the field generated a certain interest for the design of fuzzy rule base with added learning capabilities. Evolutionary computation has inspired new design of fuzzy systems with learning capabilities.

Recently, Kennedy and Eberhart [3, 4] proposed a new evolutionary computation technique, the particle swarm optimization (PSO). Due to the characters of easy implementation and rapid convergence capability, PSO has already come to be widely used in many areas. However, PSO has the problem of controlling the balance between exploration and exploitation, namely when fine tuning around the optimum is attempted.

In this paper we try to deal with this issue by introducing a novel hybrid global optimization method, termed PSODE. PSODE is based on a two-population based scheme, in which the individuals of one population is enhanced by PSO and the

individuals of the other population is evolved by DE. The interactions between the two populations influence the balance between exploration and exploitation and maintain some diversity in the whole population, even when it is approaching convergence, thus reducing the risk of convergence to local sub-optima. To demonstrate the performance of the proposed algorithm, PSODE is applied into the identification and of a typical nonlinear dynamical system, and the performance is also compared to other methods.

The paper is organized as follows. Section 2 gives a brief description of the PSO and DE. Section 3 describes the hybrid learning algorithm PSODE. Section 4 describes the T-S model and the key design procedure of fuzzy model by PSODE. In Section 5, simulation results of a nonlinear plant identification problem using fuzzy inference systems based on PSODE are presented. Finally, conclusions are drawn in Section 6.

2 Brief Review of PSO and DE

PSO is motivated by social behavior of organisms such as bird flocking and fish schooling. In PSO, the potential solutions, called particles, fly in a D -dimension search space with a velocity which is dynamically adjusted according to its own experience and that of its neighbors.

Differential evolution is a relatively new member of the evolutionary computation family. DE technique combines simple arithmetic operators with the classical events of crossover, mutation and selection to evolve from a randomly generated starting population to a final solution. The details of PSO and DE could be referred to ref. [3] and ref.[5], respectively.

3 PSODE Algorithm

In the basic PSO, all individuals are attracted by the best position found by themselves and the whole population. In this way the sharing of information among individuals is only achieved by employing the publicly available information P_g . Therefore, the population may lose diversity and is more likely to confine the search around local minima.

To solve the problem of diversity lose and premature convergence in the basic PSO model, we proposed a hybrid global optimization model. Our approach generates two population offspring individuals, one generated by the PSO mechanism and the other by DE one. In this way each individual in PSO adjusts its trajectory by three factors: its own experience, the experience of its neighbors and the experience of the individuals in the DE population. There is mutual exchange of best individual information between two populations. The idea behind the proposed algorithm is that the information can be transferred among individuals of different population that will help the individuals to avoid misjudging information and becoming trapped by poor local minima.

- (1) **Initialization.** In this step, initialize a populations with P individuals and then randomly divide them into two groups: P_1 and P_2 . These individuals could be regarded as members in terms of DE and particles in terms of PSO. In PSO, the velocities of each individuals should also be initialized

- (2) **DE operator.** Update the individuals in P_1 according to DE operations. In this paper *DE/rand/1/exp* scheme is used [6]. The new offspring can be obtained by three operations: mutation, crossover and selection.
- (3) **PSO operator.** Update the individuals in P_2 according to PSO operations, i.e. compute the position and velocity of each individual based on the PSO velocity and position equations. To further improve the performance of PSO in PSODE model, a nonlinear inertia weight approach is presented. The new inertia weight w is computed according to the following equations

$$w = w_{\min} + (w_{\max} - w_{\min}) \times \exp(-\lambda * (\frac{iter}{iter_{\max}})^2) \tag{1}$$

where, λ is control coefficient normally set as 3, $iter_{\max}$ is the maximum number of iterations, and $iter$ is the current number of iterations, w_{\max} and w_{\min} are the maximum and minimum value of inertia weight w .

- (4) **Information sharing operator.** Select the fittest local individual from P_1 and P_2 , respectively and determine the global best in the whole population. In the next generation this new fittest local individual will be used to guide the update of the individuals in P_1 and P_2 .
- (5) **Mutation operator.** If an individual (in either population) gets stagnant at any point in the search space, i.e. its location does not change for a predefined number of iterations, then the individuals is shifted by a random mutation to a new location in the search space. This technique helps escape local minima.

$$\begin{aligned} & \text{IF } (x_i^t = x_i^{t+1} = x_i^{t+2} = \dots = x_i^{t+P}) \ \& \ (F(x_i^{t+P}) \neq F^*) \\ & \quad x_i^{t+P+1} = X_{\min} + \text{rand}(0,1) * (X_{\max} - X_{\min}) \end{aligned} \tag{2}$$

END

where F^* is the global minimum of the fitness function, G is the maximum number of iterations up to which stagnation can be tolerated and $((X_{\min}, X_{\max}))$ define the permissible bounds of the search space.

- (6) **Termination.** If a terminate-condition is met, output the best performing individual of the whole population, otherwise go to step 2.

4 Fuzzy Modeling Strategy Based on the PSODE

4.1 T-S Fuzzy Model

The 1st-order T-S fuzzy models consist of linguistic IF-THEN rules which can be represented by the following general form:

$$R^i : \text{if } x_1 \text{ is } A_1^i(x_1) \text{ and } \dots x_k \text{ is } A_k^i(x_k), \text{ then } y^i \text{ is } \xi_0^i + \xi_1^i x_1 + \dots + \xi_k^i x_k, \tag{3}$$

where $A_j^i(x_j)$ are the fuzzy variables defined as follows:

$$A_j^i(x_j) = \exp[-\frac{1}{2} * [(x_j - m_j^i) / \sigma_j^i]^2], \tag{4}$$

where m_j^i and σ_j^i are the mean value and the standard deviation of the Gaussian type membership function, respectively. The details of fuzzy reasoning procedure could be referred to ref. [7].

4.2 Fuzzy Modeling Strategy Based on PSODE

Upon adopting PSODE to design a T-S fuzzy model, two key problems must be resolved, namely parameter representation and designing the fitness function.

A. Parameter representation

In our work, the parameter matrix, which consists of the premise parameters and the consequent parameters described in section 3.1, is defined as a two dimensional matrix, i.e.,

$$\begin{bmatrix} m_1^1 & \sigma_1^1 & \dots & m_n^1 & \sigma_n^1 & \alpha_0^1 & \alpha_1^1 & \dots & \alpha_n^1 \\ m_1^2 & \sigma_1^2 & \dots & m_n^2 & \sigma_n^2 & \alpha_0^2 & \alpha_1^2 & \dots & \alpha_n^2 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ m_1^r & \sigma_1^r & \dots & m_n^r & \sigma_n^r & \alpha_0^r & \alpha_1^r & \dots & \alpha_n^r \end{bmatrix}$$

The size of the matrix can be represented by $D = r \times (3n + 1)$. In particular, each individual in PSODE contains a set of premise parameters and the consequent parameters. The dimension of each individual is same as the size of the above matrix, i.e. D .

B. Fitness function designing

For each individual in PSODE, evaluate the desired optimization fitness function in D variables. The fitness function is defined as the reciprocal of RMSE (root mean quadratic error), which is used to evaluate various individuals within a population of potential solutions. Considering the single output case for clarity, our goal is to minimize the error function:

$$RMSE = \sqrt{\frac{1}{K} \sum_{k=1}^K (y_p(k+1) - y_r(k+1))^2} \tag{5}$$

where K is the total time steps, $y_p(k+1)$ is the inferred output and $y_r(k+1)$ is the desired reference output.

5 Experimental Study

In this section, simulation results of a nonlinear plant identification problem based on PSODE are presented. The plant to be identified is assumed to be of the form [7]:

$$y_p(k+1) = f[y_p(k), y_p(k-1), y_p(k-2), u(k), u(k-1)], \tag{6}$$

where the unknown function f has the form

$$f[x_1, x_2, x_3, x_4, x_5] = \frac{x_1 x_2 x_3 x_5 (x_3 - 1) + x_4}{1 + x_3^2 + x_2^2}. \tag{7}$$

In applying PSODE, 80 individuals are initially randomly generated in a population. The number r of the fuzzy rules is set to be 4; The parameters used for PSO are recommended from Shi and Eberhart [3, 4]. The maximum velocity v_{\max} and minimum velocity v_{\min} for SPSO were set at half value of the upper bound and lower bound, respectively. The acceleration constants c_1 and c_2 for PSO are both set as 2.0, the inertia weights are set to be $w_{\max} = 0.9$ and $w_{\min} = 0.4$. The DE parameters used here are $F = 0.8$ and $CR = 0.5$. For PSODE, the parameters, $c_1, c_2, w_{\max}, w_{\min}, F$ and CR , are all the same with those defined in PSO and DE. In addition, the predefined number of iterations G that stagnation can be tolerated is set as 20. This training process is shown in Fig.1.

To check the identified result, the following input as used in [8] is adopted for test:

$$\begin{aligned} u(k) &= \sin(\pi k / 25), \quad k < 250 \\ &= 1.0, \quad 250 \leq k < 500 \\ &= -1.0, \quad 500 \leq k < 750 \\ &= 0.3 \sin(\pi k / 25) + 0.1 \sin(\pi k / 32) + 0.6 \sin(\pi k / 10), \quad 750 \leq k < 1000. \end{aligned}$$

For fair comparison, the methods for the plant identification use the same train data set and test data set, and the train time steps are all equals to 900. Fig.2 shows the desired output and the inferred output obtained by the fuzzy rule base evolve by PSODE, where the solid curve represents the desired output and the dotted curve represent the actual output.

Table 1. Performance comparisons with different methods

Method	RFNN	TRFN-S	PSO	DE	PSODE
RMSE(train)	0.0114	0.0084	0.0386	0.0552	0.0346
RMSE (test)	0.0575	0.0346	0.0372	0.0424	0.0184

Table 1 gives the identification results of the nonlinear plant using different methods, where the results of the methods RFNN and TRFN-S come from literature [9]. From Table 1, we can see that although the fuzzy controller designed by PSODE can not outperform the methods using RFNN and TRFN-S, it achieves the highest identification accuracy in the test part, which demonstrate its better generalized ability. The results of PSODE identifier also demonstrate the performance improved by the hybrid of the PSO and DE compared to the results of the identifiers obtained independently using PSO or DE.

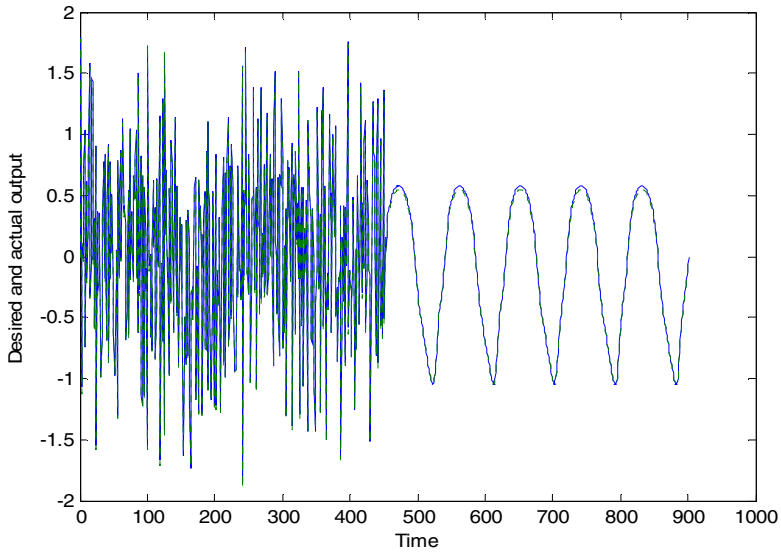


Fig. 1. Training process of PSODE based T-S fuzzy model

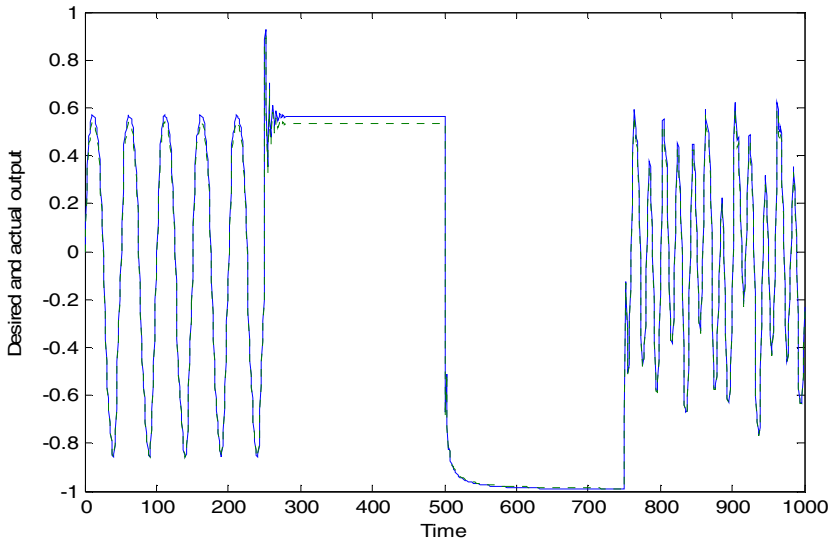


Fig. 2. Identification results using PSODE

6 Conclusions and Future Work

A new design approach for T-S fuzzy model is presented based on the hybrid particle swarm optimization and differential evolution. The new learning algorithm in the T-S

model is named PSONE. PSONE combines the merits of PSO and DE, in which PSO and DE are executed in parallel with a frequent information sharing. In this way, the information can be transferred among individuals between PSO and DE that will help the individuals to avoid misjudging information and becoming trapped by poor local minima. Furthermore a nonlinear inertia weight approach and a mutation mechanism are embedded into PSONE to further improve the performance of PSONE..

In order to demonstrate the performance of the proposed T-S fuzzy model, it is applied to the identification of nonlinear dynamical systems. The experimental results show that the identification performance greatly outperforms several typical methods in dynamical system.

Acknowledgements. This work is supported by the National Natural Science Foundation of China under Grant No. 70271014.

References

1. Tseng, C.S., Chen, B.S., Uang, H.J.: Fuzzy Tracking Control Design for Nonlinear Dynamical Systems via T-S Fuzzy Model. *IEEE Trans. Fuzzy Syst.* 9, 81–392 (2001)
2. Wang, H.O., Tanak, K.A., Griffin, M.F.: An Approach to Fuzzy Control of Nonlinear Systems: Stability and Design Issues. *IEEE Trans. Fuzzy Syst.* 4, 14–23 (1996)
3. Eberchart, R.C., Kennedy, J.: A New Optimizer Using Particle Swarm Theory. In: 6th IEEE International Symposium on Micromachine and Human Science, pp. 39–43. IEEE Press, Piscataway (1995)
4. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: Proceeding of IEEE International Conference on Neural Networks, pp. 1942–1948. IEEE Press, Piscataway (1995)
5. Price, K.V.: An Introduction to Differential Evolution. In: Corne, D., Dorigo, M., Glover, F. (eds.) *New Ideas in Optimization*, pp. 79–108. McGraw-Hill, London (1999)
6. Storn, R., Price, K.V.: Differential Evolution: A Simple and Efficient Heuristic Strategy for Global Optimization over Continuous Spaces. *Journal of Global Optimization* 11, 341–359 (1997)
7. Narendra, K.K., Parthasarathy, K.: Identification and Control of Dynamical Systems Using Neural Networks. *IEEE Trans. Neural Network* 1, 4–27 (1990)
8. Juang, C.F.: A TSK-type Recurrent Fuzzy Network for Dynamic Systems Processing by Neural Network and Genetic Algorithm. *IEEE Trans. Fuzzy Syst.* 10, 155–170 (2002)
9. Lee, C.H., Teng, C.C.: Identification and Control of Dynamic Systems Using Recurrent Fuzzy Neural Networks. *IEEE Trans. Fuzzy Syst.* 8, 349–366 (2000)

Equivalence Knowledge Mass and Approximate Reasoning in \mathcal{R} -Logic $\mathbb{C}_{\mathcal{R}}$ (I)

Yalin Zheng^{1,2}, Guang Yang¹, Changshui Zhang²,
Jing Zheng^{3,1}, and Yunpeng Xu²

¹ Research Center of Intelligent Information Technology,
Department of Computer Science and Technology, Faculty of Software,
Dongguan University of Science and Technology, Dongguan 523808, China

² State Key Laboratory for Intelligent Technology and Systems,
Department of Automation, Faculty of Information Science and Technology,
Tsinghua University, Beijing 100084, China

³ Institute of Dongguan Economy and International Commerce,
Department of Economy and Commerce,
Dongguan University of Science and Technology, Dongguan 523808, China
zheng-yl@mail.tsinghua.edu.cn, bgzhengyl@dgut.edu.cn

Abstract. By casting off the direct restriction of topological structure, this paper presents another matching scheme between the input A^* and the knowledge $A \rightarrow B$ based on the equivalence relation \mathcal{R} on formulae set $\mathcal{F}(\mathcal{S})$ and the corresponding equivalence classification

$$\mathcal{F}(\mathcal{S})/\mathcal{R} = \{[A]_{\mathcal{R}} \mid A \in \mathcal{F}(\mathcal{S})\}$$

therefore, obtains another algorithm of approximate reasoning — the IV-type \mathcal{R} -algorithm.

Keywords: Knowledge Cumularspherolith, Formulae Cumularspherolith, Approximate Knowledge Closure of Knowledge Base, Pseudodistance, \mathcal{R} -automatic Reasoning System, \mathcal{R} -completeness.

1 Introduction

Many logicians have done a lot of research work for providing the logical foundation and logical normalization of the approximate reasoning. Guo-Jun Wang constructed a fuzzy propositional logic system \mathcal{L}^* and gave the $\alpha-3I$ algorithm of the approximate reasoning in [1]. In [2], Wang studied the logical metric space based on their integrate semantic theory. The approximate reasoning model in the framework of the classical propositional logic is also studied in [3]. Mingsheng Ying gave a logic model of approximate reasoning in [4] and studied approximate reasoning based on the fuzzy matching in [5]. We constructed the framework of the stratified fuzzy propositional logic in [6] and discovered an important logical property of Mamdani algorithm for fuzzy reasoning in [7]. All the above work motivate the authors to study the approximate reasoning model in the framework of topological logic [8][9][10][11].

The key problem of the approximate reasoning that AI theories and techniques rely on is, how to approximate the matching between the input A^* and the knowledge $A \rightarrow B$. As stated in [8-10], we have given three different kinds of schemes according to the different topological logic structures and their corresponding construction of the matching functions, thus we have given three different topological algorithms of the approximate reasoning, all of which agree in a certain sense with the common senses of human’s approximate reasoning. Recall these three algorithms, the key idea is based on such an understanding, that is, the topological structure can score rightly the similarity degree between one thing and another, therefore, it can be used to describe and regulate how the input A^* and the knowledge $A \rightarrow B$ match. However, when we get to the bottom, we shall cognize at a higher level that, the essential of the topological algorithms and matching schemes we established is a group constituted by the following three *formulae cumularspharolithes*

$$(U_A, U_B, U_{A \rightarrow B}),$$

which are specified for each implication $A \rightarrow B$.

These formulae cumularspharolithes

$$(U_A, U_B, U_{A \rightarrow B})$$

can be described with either topological method, as we have done before, or some other methods, as we will.

In this paper, we will cast off the direct restriction by topological structure, based on the equivalence relation \mathcal{R} on formulae set $\mathcal{F}(\mathcal{S})$ and the corresponding equivalence classification

$$\mathcal{F}(\mathcal{S})/\mathcal{R} = \{[A]_{\mathcal{R}} \mid A \in \mathcal{F}(\mathcal{S})\}$$

we propose another scheme of matching between the input A^* and the knowledge $A \rightarrow B$ — the IV-type matching scheme, therefore, obtain another algorithm of approximate reasoning — the IV-type \mathcal{R} –algorithm. According to this viewpoint, the equivalence replacement theorem in the classical proposition logic \mathbb{C} becomes a special case of our IV-type \mathcal{R} –algorithm for approximate reasoning.

2 Equivalence Relation \mathcal{R} , Pseudo-distance $d_{\mathcal{R}}$, Topology $\mathcal{I}_{\mathcal{R}}$ on Formulae Set $\mathcal{F}(\mathcal{S})$ and Construction of \mathcal{R} –Logic $\mathbb{C}_{\mathcal{R}}$

In classical propositional logic $\mathbb{C} = (\widehat{\mathbb{C}}, \widetilde{\mathbb{C}})$, let \mathcal{R} be an arbitrary equivalence relation on formulae set $\mathcal{F}(\mathcal{S})$, such that each implication $A \rightarrow B \in \mathcal{H}$ satisfies

$$[A]_{\mathcal{R}} \rightarrow [B]_{\mathcal{R}} \subseteq [A \rightarrow B]_{\mathcal{R}}$$

here,

$$[A]_{\mathcal{R}} \rightarrow [B]_{\mathcal{R}} = \{A^* \rightarrow B^* \mid A^* \in [A]_{\mathcal{R}}, B^* \in [B]_{\mathcal{R}}\}$$

$(\mathcal{F}(\mathcal{S}), \mathcal{R})$ is termed \mathcal{R} -space, we call

$$\mathbb{C}_{\mathcal{R}} = (\widehat{\mathbb{C}}, \widetilde{\mathbb{C}}, \mathcal{R})$$

\mathcal{R} -logic.

Example 2.1. In formulae set $\mathcal{F}(\mathcal{S})$, consider following equivalence relations

- (1) Identity relation \mathcal{R} ;
- (2) Provable equivalence relation \mathcal{R} ;
- (3) Logical equivalence relation \mathcal{R} ;
- (4) The logical equivalence relation \mathcal{R} in terms of partly evaluations ;
- (5) The equivalence relation \mathcal{R} in terms of zero measurement under the all evaluations ;
- (6) The equivalence relation \mathcal{R} in terms of general measurement under the all evaluations ;
- (7) The equivalence relation \mathcal{R} in terms of zero measurement under the partly evaluations ;
- (8) The equivalence relation \mathcal{R} in terms of general measurement under the partly evaluations ;

Then $(\mathcal{F}(\mathcal{S}), \mathcal{R})$ constructs a \mathcal{R} -space, and $\mathbb{C}_{\mathcal{R}} = (\widehat{\mathbb{C}}, \widetilde{\mathbb{C}}, \mathcal{R})$ constructs a \mathcal{R} -logic.

The quotient set of $\mathcal{F}(\mathcal{S})$ with respect to the equivalence relation \mathcal{R} is

$$\mathcal{F}(\mathcal{S})/\mathcal{R} = \{[A]_{\mathcal{R}} \mid A \in \mathcal{F}(\mathcal{S})\}$$

It forms a partition of the formulae set $\mathcal{F}(\mathcal{S})$, i.e.,

- (1) $\mathcal{F}(\mathcal{S})/\mathcal{R}$ covers $\mathcal{F}(\mathcal{S})$,

$$\bigcup_{[A]_{\mathcal{R}} \in \mathcal{F}(\mathcal{S})/\mathcal{R}} [A]_{\mathcal{R}} = \mathcal{F}(\mathcal{S})$$

- (2) The elements in $\mathcal{F}(\mathcal{S})/\mathcal{R}$ are pairwise disjoint,

$$[A]_{\mathcal{R}} \neq [B]_{\mathcal{R}} \Leftrightarrow [A]_{\mathcal{R}} \cap [B]_{\mathcal{R}} = \emptyset$$

Per contra, any partition \mathcal{P} of formulae set $\mathcal{F}(\mathcal{S})$, i.e., the subset family that satisfies the following two items

- (1) $\bigcup_{U \in \mathcal{P}} U = \mathcal{F}(\mathcal{S})$
- (2) $U, V \in \mathcal{P}, U \neq V \Leftrightarrow U \cap V = \emptyset$

determines an equivalence relation $\mathcal{R}_{\mathcal{P}}$ on $\mathcal{F}(\mathcal{S})$, therefore, it's suffice to let

$$A \mathcal{R}_{\mathcal{P}} B \text{ iff there exists } U \in \mathcal{P} \text{ such that } A, B \in U$$

and the quotient set $\mathcal{F}(\mathcal{S})/\mathcal{R}_{\mathcal{P}}$ determined by the equivalence relation $\mathcal{R}_{\mathcal{P}}$ superposes the partition \mathcal{P} , i.e.,

$$\mathcal{F}(\mathcal{S})/\mathcal{R}_{\mathcal{P}} = \mathcal{P}$$

In \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$, for each \mathcal{R} -equivalence class $[A]_{\mathcal{R}} \in \mathcal{F}(\mathcal{S})/\mathcal{R}$, $[A]_{\mathcal{R}}$ is also called \mathcal{R} -*formulae cumularspherolith* with the *core* of formula A , or simply \mathcal{R} -*cu- mularspherolith*. In particular, for each theorem $A \rightarrow B \in \mathcal{O}^+$, the \mathcal{R} -equivalence class $[A \rightarrow B]_{\mathcal{R}}$ is called \mathcal{R} -*knowledge cumularspherolith* with the *core* of theorem $A \rightarrow B$.

Theorem 2.1. In \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$, define mapping

$$d_{\mathcal{R}} : \begin{cases} \mathcal{F}(\mathcal{S}) \times \mathcal{F}(\mathcal{S}) \rightarrow [0, +\infty) \\ (A, B) \mapsto d_{\mathcal{R}}(A, B) = \begin{cases} 0, [A]_{\mathcal{R}} = [B]_{\mathcal{R}} \\ 1, [A]_{\mathcal{R}} \cap [B]_{\mathcal{R}} = \emptyset \end{cases} \end{cases}$$

then $d_{\mathcal{R}}$ is a *pseudo-distance* on formulae set $\mathcal{F}(\mathcal{S})$, therefore, $(\mathcal{F}(\mathcal{S}), d_{\mathcal{R}})$ forms a *pseudo-distance space*. We call $d_{\mathcal{R}}$ the *pseudo-distance induced by the equivalence relation \mathcal{R}* , and $(\mathcal{F}(\mathcal{S}), d_{\mathcal{R}})$ the *pseudo-distance space induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$* .

Theorem 2.2. Let

$$d : \begin{cases} \mathcal{F}(\mathcal{S}) \times \mathcal{F}(\mathcal{S}) \rightarrow [0, +\infty) \\ (A, B) \mapsto d(A, B) = 0 \text{ or } 1 \end{cases}$$

be a pseudo-distance on formulae set $\mathcal{F}(\mathcal{S})$, we define a binary relation on $\mathcal{F}(\mathcal{S})$ as follows

$$A\mathcal{R}_d B \Leftrightarrow d(A, B) = 0$$

Then \mathcal{R}_d is an equivalence relation on formulae set $\mathcal{F}(\mathcal{S})$, termed the *equivalence relation induced by pseudo-distance d* .

Theorem 2.3. Let

$$d : \begin{cases} \mathcal{F}(\mathcal{S}) \times \mathcal{F}(\mathcal{S}) \rightarrow [0, +\infty) \\ (A, B) \mapsto d(A, B) = 0 \text{ or } 1 \end{cases}$$

be a pseudo-distance on formulae set $\mathcal{F}(\mathcal{S})$, \mathcal{R}_d be the equivalence relation induced by the pseudo-distance d , $d_{\mathcal{R}_d}$ is the pseudo-distance induced by the equivalence relation \mathcal{R}_d , then

$$d_{\mathcal{R}_d} = d \tag{1}$$

Theorem 2.4. Let \mathcal{R} be an equivalence relation on formulae set $\mathcal{F}(\mathcal{S})$, $d_{\mathcal{R}}$ be the pseudo-distance induced by the equivalence relation \mathcal{R} , $\mathcal{R}_{d_{\mathcal{R}}}$ be the equivalence induced by the pseudo-distance $d_{\mathcal{R}}$, then

$$\mathcal{R}_{d_{\mathcal{R}}} = \mathcal{R} \tag{2}$$

We endow the formulae set $\mathcal{F}(\mathcal{S})$ in classical proposition logic $\mathbb{C} = (\widehat{\mathbb{C}}, \widetilde{\mathbb{C}})$ with a pseudo-distance

$$d : \begin{cases} \mathcal{F}(\mathcal{S}) \times \mathcal{F}(\mathcal{S}) \rightarrow [0, +\infty) \\ (A, B) \mapsto d(A, B) = 0 \text{ or } 1 \end{cases}$$

such that any $A, B, C \in \mathcal{F}(\mathcal{S})$ satisfies

$$d(A^*, A) = d(B^*, B) = 0 \Rightarrow d(A^* \rightarrow B^*, A \rightarrow B) = 0$$

i.e., when both the pseudo-distances between A^* and A and between B^* and B are 0, the pseudo-distance between the implications $A^* \rightarrow B^*$ and $A \rightarrow B$ must also be 0, then $(\mathcal{F}(\mathcal{S}), d)$ is termed d -space. We call

$$\mathbb{C}_d = (\widehat{\mathbb{C}}, \widetilde{\mathbb{C}}, d)$$

d -logic.

Theorem 2.5. Let $d_{\mathcal{R}}$ be the pseudo-distance induced by the equivalence relation \mathcal{R} in \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$, then for any formulae $A, B, A^*, B^* \in \mathcal{F}(\mathcal{S})$, $d_{\mathcal{R}}$ must satisfy

$$\begin{aligned} d_{\mathcal{R}}(A^*, A) = d_{\mathcal{R}}(B^*, B) = 0 \Rightarrow \\ d_{\mathcal{R}}(A^* \rightarrow B^*, A \rightarrow B) = 0 \end{aligned}$$

i.e., when both the pseudo-distances between A^* and A and between B^* and B are 0, then the pseudo-distance between the implications $A^* \rightarrow B^*$ and $A \rightarrow B$ must also be 0, therefore

$$(\mathcal{F}(\mathcal{S}), d_{\mathcal{R}})$$

forms a $d_{\mathcal{R}}$ -space, called $d_{\mathcal{R}}$ -space induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$;

$$\mathbb{C}_{d_{\mathcal{R}}} = (\widehat{\mathbb{C}}, \widetilde{\mathbb{C}}, d_{\mathcal{R}})$$

forms a $d_{\mathcal{R}}$ -logic, called $d_{\mathcal{R}}$ -logic generated by \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$.

Theorem 2.6. Let \mathcal{R}_d be the equivalence relation induced by the pseudo-distance d in d -logic \mathbb{C}_d , then for any formulae $A, B, \in \mathcal{F}(\mathcal{S})$, \mathcal{R}_d must satisfy

$$[A]_{\mathcal{R}_d} \rightarrow [B]_{\mathcal{R}_d} \subseteq [A \rightarrow B]_{\mathcal{R}_d}$$

Here,

$$[A]_{\mathcal{R}_d} \rightarrow [B]_{\mathcal{R}_d} = \{A^* \rightarrow B^* \mid A^* \in [A]_{\mathcal{R}_d}, B^* \in [B]_{\mathcal{R}_d}\}$$

Therefore,

$$(\mathcal{F}(\mathcal{S}), \mathcal{R}_d)$$

forms a \mathcal{R}_d -space, called \mathcal{R}_d -space induced by d -space $(\mathcal{F}(\mathcal{S}), d)$;

$$\mathbb{C}_{\mathcal{R}_d} = (\widehat{\mathbb{C}}, \widetilde{\mathbb{C}}, \mathcal{R}_d)$$

forms a \mathcal{R}_d -logic, called \mathcal{R}_d -logic induced by d -logic \mathbb{C}_d .

Theorem 2.7. Let $\mathbb{C}_{\mathcal{R}}$ be \mathcal{R} -logic, $\mathbb{C}_{d_{\mathcal{R}}}$ be the $d_{\mathcal{R}}$ -logic induced by \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$, $\mathbb{C}_{\mathcal{R}_{d_{\mathcal{R}}}}$ be the $\mathcal{R}_{d_{\mathcal{R}}}$ -logic induced by $d_{\mathcal{R}}$ -logic $\mathbb{C}_{d_{\mathcal{R}}}$, then

$$\mathbb{C}_{\mathcal{R}_{d_{\mathcal{R}}}} = \mathbb{C}_{\mathcal{R}}$$

Theorem 2.8. Let \mathbb{C}_d be d -logic, $\mathbb{C}_{\mathcal{R}_d}$ be the \mathcal{R}_d -logic induced by d -logic \mathbb{C}_d , $\mathbb{C}_{d_{\mathcal{R}_d}}$ be the $d_{\mathcal{R}_d}$ -logic induced by \mathcal{R}_d -logic $\mathbb{C}_{\mathcal{R}_d}$, then

$$\mathbb{C}_{d_{\mathcal{R}_d}} = \mathbb{C}_d$$

Theorem 2.9. In \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$, it is definite to use the quotient set of $\mathcal{F}(\mathcal{S})$ with respect to the equivalence relation \mathcal{R}

$$\mathcal{F}(\mathcal{S})/\mathcal{R} = \{[A]_{\mathcal{R}} \mid A \in \mathcal{F}(\mathcal{S})\}$$

as the base to generate a topology

$$\mathcal{T}_{\mathcal{R}} = \{U \mid \exists \mathcal{U} \subseteq \mathcal{F}(\mathcal{S})/\mathcal{R}, \text{ s.t. } U = \bigcup_{[A]_{\mathcal{R}} \in \mathcal{U}} [A]_{\mathcal{R}}\}$$

Topology $\mathcal{T}_{\mathcal{R}}$ is called the *topology induced by the equivalence relation \mathcal{R}* , and $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ is called the *topological space induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$* .

Theorem 2.10. In the topological space $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$, let $\mathcal{F}_{\mathcal{R}}$ denote the family of all closed sets, then

$$\mathcal{F}_{\mathcal{R}} = \mathcal{T}_{\mathcal{R}}$$

This indicates that a set in this space is a closed set, if and only if it is an open set, therefore, topology $\mathcal{T}_{\mathcal{R}}$ or the closed sets family $\mathcal{F}_{\mathcal{R}}$ is exactly the family of all the both open and closed sets in this space.

In the topological space $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$, for each subset $\mathcal{A} \subseteq \mathcal{F}(\mathcal{S})$ of the formulae set $\mathcal{F}(\mathcal{S})$, \mathcal{A}^o and \mathcal{A}^- denote the *interior* and the *closure* of \mathcal{A} , respectively, then

$$\begin{aligned} \mathcal{A}^o &= \bigcup \{\mathcal{G} \mid \mathcal{G} \in \mathcal{T}_{\mathcal{R}}, \mathcal{G} \subseteq \mathcal{A}\} \\ \mathcal{A}^- &= \bigcap \{\mathcal{G} \mid \mathcal{G} \in \mathcal{F}_{\mathcal{R}}, \mathcal{A} \subseteq \mathcal{G}\} \end{aligned}$$

i.e., \mathcal{A}^o , the interior of \mathcal{A} , is exactly the largest open set contained by \mathcal{A} , while \mathcal{A}^- , the closure of \mathcal{A} , is exactly the smallest closed set contains \mathcal{A} . In the special topological space $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$, \mathcal{A}^o is also called the *inferior approximation* of \mathcal{A} , while \mathcal{A}^- is the *superior approximation* of \mathcal{A} . For the subset \mathcal{A} of formulae set $\mathcal{F}(\mathcal{S})$, if \mathcal{A}^o superposes \mathcal{A}^- , i.e.,

$$\mathcal{A}^o = \mathcal{A} = \mathcal{A}^-$$

then we call \mathcal{A} a *definable set* on $\mathcal{F}(\mathcal{S})$. If \mathcal{A}^o does not superpose \mathcal{A}^- , i.e.,

$$\mathcal{A}^o \subset \mathcal{A} \subset \mathcal{A}^-$$

then we call \mathcal{A} a *rough set* on $\mathcal{F}(\mathcal{S})$. Therefore, a set \mathcal{A} is a definable set, if and only if \mathcal{A} is a both open and closed set; a set \mathcal{A} is a rough set, if and only if \mathcal{A} is not a both open and closed set.

Theorem 2.11. In the topological space $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$, for each formula $A \in \mathcal{F}(\mathcal{S})$, let $\mathcal{U}^o(A)$ denote the open neighborhood system of A , i.e.,

$$\mathcal{U}^o(A) = \{V \mid A \in V \in \mathcal{T}_{\mathcal{R}}\}$$

and $\mathcal{U}(A)$ denote the neighborhood system of A , then

$$\mathcal{U}(A) = \{U \mid \exists V \in \mathcal{U}^o(A), \text{ s.t. } V \subseteq U\}$$

and all the open neighborhoods in $\mathcal{U}^o(A)$ are definable set, or termed the *definable neighborhood* of A ; all the neighborhoods in $\mathcal{U}(A) - \mathcal{U}^o(A)$ are rough set, or termed *rough neighborhood* of A .

Theorem 2.12. In the topological space $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$, for each formula $A \in \mathcal{F}(\mathcal{S})$,

$$\mathcal{B}(A) = \mathcal{U}^o(A) \cap (\mathcal{F}(\mathcal{S})/\mathcal{R}) = \{[A]_{\mathcal{R}}\}$$

forms the simplest neighborhood base of formula A , thus the topological space $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ is the first countable space.

Theorem 2.13. In the topological space $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$, for any formula $A \in \mathcal{F}(\mathcal{S})$, there exists the smallest open neighborhood $[A]_{\mathcal{R}}$, i.e.,

$$\min \mathcal{U}(A) = \min \mathcal{U}^o(A) = \min \mathcal{B}(A) = [A]_{\mathcal{R}}$$

therefore,

$$\bigcap_{U \in \mathcal{U}(A)} U = \bigcap_{V \in \mathcal{U}^o(A)} V = \bigcap_{W \in \mathcal{B}(A)} W = [A]_{\mathcal{R}}$$

Theorem 2.14. In the topological space $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$ in \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$, for any formulae $A, B \in \mathcal{F}(\mathcal{S})$

$$\begin{aligned} \{A\}^- &= [A]_{\mathcal{R}}, \{B\}^- = [B]_{\mathcal{R}} \\ \{A \rightarrow B\}^- &= [A \rightarrow B]_{\mathcal{R}} \\ \{A\}^o &= \emptyset, \{B\}^o = \emptyset \\ \{A \rightarrow B\}^o &= \emptyset \end{aligned}$$

Theorem 2.15. In the topological space $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}),$

$\mathcal{R})$ in \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$, for any formulae $A, B \in \mathcal{F}(\mathcal{S})$ the topological closure operator $-$ must satisfy

$$\{A\}^- \rightarrow \{B\}^- \subseteq \{A \rightarrow B\}^-$$

therefore, logic \mathbb{C} forms the I-type topological logic by topology $\mathcal{T}_{\mathcal{R}}$

$$\mathbb{C}_{\mathcal{T}_{\mathcal{R}}}^I = (\widehat{\mathbb{C}}, \widetilde{\mathbb{C}}, \mathcal{T}_{\mathcal{R}})$$

Theorem 2.16. In the topological space $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$ in \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$, for each implication $A \rightarrow B \in \mathcal{H}$,

$$\begin{aligned} \forall W \in \mathcal{U}(A \rightarrow B), \exists U \in \mathcal{U}(A), \\ \exists V \in \mathcal{U}(B), \text{ s.t. } U \rightarrow V \subseteq W. \end{aligned}$$

Therefore, logic \mathbb{C} forms II-type topological logic by topology $\mathcal{T}_{\mathcal{R}}$

$$\mathbb{C}_{\mathcal{T}_{\mathcal{R}}}^{II} = (\widehat{\mathbb{C}}, \widetilde{\mathbb{C}}, \mathcal{T}_{\mathcal{R}})$$

Theorem 2.17. In the topological space $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$ in \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$, define selection function

$$\mathcal{C} : \begin{cases} \mathcal{H} \rightarrow \bigcup_{A \rightarrow B \in \mathcal{H}} (\mathcal{U}(A) \times \mathcal{U}(B) \times \mathcal{U}(A \rightarrow B)) \\ (A \rightarrow B) \mapsto \mathcal{C}(A \rightarrow B) = ([A]_{\mathcal{R}}, [B]_{\mathcal{R}}, [A \rightarrow B]_{\mathcal{R}}) \end{cases}$$

then \mathcal{C} is the II-type matching function on \mathcal{H} ,

$$([A]_{\mathcal{R}}, [B]_{\mathcal{R}}, [A \rightarrow B]_{\mathcal{R}})$$

is the \mathcal{C} -matching neighborhood group with respect to the implication $A \rightarrow B$.

Theorem 2.18. Let $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ be the topological space induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$ in \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$, then logic \mathbb{C} forms III-type topological logic

$$\mathbb{C}_{\mathcal{T}_{\mathcal{R}}}^{III} = (\widehat{\mathbb{C}}, \widetilde{\mathbb{C}}, \mathcal{T}_{\mathcal{R}})$$

on topology $\mathcal{T}_{\mathcal{R}}$.

Theorem 2.19. In the topological space $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$ in \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$, define mapping \mathcal{D} , such that for each implication $A \rightarrow B \in \mathcal{H}$,

$$\mathcal{D}(A \rightarrow B) = (\mathcal{B}_A, \mathcal{B}_B, \mathcal{B}_{A \rightarrow B})$$

Here,

$$\begin{aligned} \mathcal{B}_A &= \mathcal{B}(A) = \{[A]_{\mathcal{R}}\} \\ \mathcal{B}_B &= \mathcal{B}(B) = \{[B]_{\mathcal{R}}\} \\ \mathcal{B}_{A \rightarrow B} &= \mathcal{B}(A \rightarrow B) = \{[A \rightarrow B]_{\mathcal{R}}\} \end{aligned}$$

Then \mathcal{D} is the III-type matching function on \mathcal{H} ,

$$(\mathcal{B}_A, \mathcal{B}_B, \mathcal{B}_{A \rightarrow B})$$

is the \mathcal{D} -matching neighborhood bundle with respect to the implication $A \rightarrow B$.

Theorem 2.20. Let $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ be the topological space induced by \mathcal{R} -space $(\mathcal{F}(\mathcal{S}), \mathcal{R})$ in \mathcal{R} -logic $\mathbb{C}_{\mathcal{R}}$, then if the equivalence relation \mathcal{R} on formulae set $\mathcal{F}(\mathcal{S})$ is not an identity relation, topology $\mathcal{T}_{\mathcal{R}}$ cannot be a Fréchet topology, even not a Колмогоров topology.

Theorem 2.20 actually tells us that, in the case of rough set, a topology constructed by the entire definable sets cannot be a Колмогоров topology, let alone a Fréchet topology or a Hausdorff topology.

3 Conclusion

Although we mentioned to cast off the direct restriction by topological structure, we have seen in this paper that, it is quite easy to introduce a pseudo-distance $d_{\mathcal{R}}$ on the formulae set $\mathcal{F}(\mathcal{S})$ equipped with equivalence relation \mathcal{R} , such that $(\mathcal{F}(\mathcal{S}), d_{\mathcal{R}})$ is a pseudo-distance space, therefore, it makes sense to discuss the distance and similarity degree between the formulae on $\mathcal{F}(\mathcal{S})$, then we can use the pseudo-distance and the similarity degree scored by the former to describe the IV-type \mathcal{R} -algorithm of the approximate reasoning. It is also easy to prove that, the equivalence relation \mathcal{R} and the pseudo-distance on the formulae set $\mathcal{F}(\mathcal{S})$ are equivalent and inter-determinable.

Besides, as we have seen in the paper, it is quite convenient to give a topology $\mathcal{T}_{\mathcal{R}}$ on the formulae set $\mathcal{F}(\mathcal{S})$ based on its equivalence relation \mathcal{R} and the corresponding equivalence classification $\mathcal{F}(\mathcal{S})/\mathcal{R}$, such that $(\mathcal{F}(\mathcal{S}), \mathcal{T}_{\mathcal{R}})$ is a topological space. Consequently, the IV-type \mathcal{R} -algorithm can also be described equivalently through topological structure. Therefore, the IV-type \mathcal{R} -algorithm is also brought into the orbit of topological logic and topological algorithm. We also prove, in case of rough set, a topology constructed by the entire definable sets cannot be a Колмогоров topology, let alone a Fréchet topology or a Hausdorff topology.

References

1. Wang, G.J.: On the Logic Foundation of Fuzzy Reasoning. *Information Science* 117, 47–88 (1999)
2. Wang, G.J., Leung, Y.: Intergrated Semantics and Logic Metric Spaces. *Fuzzy Sets and Systems* 136, 71–91 (2003)
3. Wang, G.J., Wang, H.: Non-fuzzy Versions of Fuzzy Reasoning in Classical Logic. *Information Sciences* 138, 211–236 (2001)
4. Ying, M.S.: A Logic for Approximate Reasoning. *The Journal of Symbolic Logic* 59, 830–837 (1994)
5. Ying, M.S.: Fuzzy Reasoning under Approximate Match. *Science Bulletin* 37, 1244–1245 (1992)
6. Zheng, Y.L.: Stratified Fuzzy Propositional Logic and Approximate Reasoning. Doctorate Dissertation of Xi'an Jiaotong University, vol. 1, pp. 1–180 (2002)
7. Zheng, Y.L.: Stratified Construction of Fuzzy Propositional Logic. In: *Proceedings of International Conference on Fuzzy Information Processing*, pp. 1–2, 169–174. Tsinghua University Press/Springer(2003)
8. Zheng, Y.L., Zhang, C.S., Yi, X.: Mamdaniean logic. In: *Proceedings of IEEE International Conference on Fuzzy Systems*, Budapest, Hungary, pp. 1–3, 629–634 (2004)
9. Zheng, Y.L., Zhang, C.S., Yao, X.: Type I Topological Logic and Approximate Reasoning. In: Wang, L., Jin, Y. (eds.) *FSKD 2005. LNCS (LNAI)*, vol. 3613, pp. 253–262. Springer, Heidelberg (2005)
10. Zheng, Y.L.: Topological Logic and Approximate Reasoning. Postdoctor Research Report of Tsinghua University, vol. 1, pp. 1–326 (2006)

11. Zheng, Y.L., Zhang, C.S., Xia, Y.L.: Type II Topological Logic and Approximate Reasoning. In: Wang, L., Jin, Y. (eds.) FSKD 2005. LNCS (LNAI), vol. 3613, pp. 243–252. Springer, Heidelberg (2005)
12. Meng, G.W.: Lowen's Compactness in L-fuzzy Topological Spaces. *Fuzzy Sets and Systems* 53, 329–333 (1993)
13. Bai, S.Z.: Q-convergence of nets and weak separation axiom in fuzzy lattices. *Fuzzy Sets and Systems* 88, 379–386 (1997)
14. Pavalka, J.: On Fuzzy Logic (I)(II)(III). *Zeitschr. F. Math. Logik und Grundlagen d. Math.* 25, 45–52, 119–134, 447–464 (1979)
15. Novak, V.: On the Syntactic-semantical Completeness of First Order Fuzzy Logic (I)(II). *Kybernetika* 26, 47–66, 134–154 (1990)
16. Thomas, S.: Similarity, Interpolation, and Fuzzy Rule Construction. *Fuzzy Sets and Systems* 58, 73–86 (1993)
17. Pappis, C.P., Nikos, I.K.: A Comparative Assessment of Measures of Similarity of Fuzzy Values. *Fuzzy Sets and Systems* 56, 171–174 (1993)
18. Pappis, C.P.: Value Approximation of Fuzzy Systems. *Fuzzy Sets and Systems* 39, 111–115 (1991)
19. Kickert, W.J.M., Mamdani, E.H.: Analysis of a Fuzzy Logic Controller. *Fuzzy Sets and Systems* 1, 29–44 (1978)
20. Czogala, E., Leski, J.: On Equivalence of Approximate Reasoning Results Using Different Interpretations of Fuzzy if-then rules. *Fuzzy Sets and Systems* 117, 279–296 (2001)
21. Amao, P., Nola, A.D., Navara, M.: Criteria That should be Satisfied by Mamdani-Assilian Controller. In: Proceedings of the International conference on Fuzzy Information Processing, pp. 1–2, 195–198. Tsinghua University Press/Springer (2003)
22. Ying, M.S.: Reasoning about Probabilistic Sequential Programs in a Probabilistic Logic. *Acta Informatica* 39, 315–389 (2003)

Extended Fuzzy Knowledge Representation with Medium

Cen Wang and Zhenghua Pan

School of Science, Jiangnan University, Wuxi, Jiangsu, China
wangcen715@163.com

Abstract. Found on theories of Medium Logic, the paper¹ differentiates contradictory from opposition in knowledge representation, affirms that there exists medium concept between a pair of opposite fuzzy concepts and introduces fuzzy privative and opposition privative in traditional knowledge representation. It takes a financial decision-making program for example, adopts ratio function of distance to endow each fuzzy predicate expression with a truth value which belongs to interval $[0, 1]$. Found on infinite-valued model of MF, it gives a method to ascertain value of controlled variable λ in model according to specific case, from which the effect and meaning of λ are shown. Finally it discusses about fuzzy knowledge reasoning by a specific example.

Keywords: Fuzzy knowledge representation, Medium Logic, infinite-valued model of MF, truth value.

1 Medium Logic System

1.1 Basic Concept

Since Aristotle, formal logic has differentiated contradictory from opposition in concept. If one concept's intension denies another concept's intension, they are called a pair of contradictory concepts. For example, *good* and *not good*. If either concept has its own positive meaning while there exists the largest difference between the two concepts in a higher level concept in the same intension, they are called a pair of opposite concepts, such as good and bad. However, not everything in nature only has positive aspect and negative aspect. Many things have medium between positive aspect and negative aspect. Medium logic is founded on this background.

In medium predicate logic, suppose P is an unitary predicate, opposition of P is written as $\neg P$. P and $\neg P$ are a pair of opposite predicates. If there exists object x , and $P(x)$ and $\neg P(x)$ are both partly true, such x is called medium object of both $P(x)$ and $\neg P(x)$. Such character the x has is written as $\sim P$ which is called medium predicate of both P and $\neg P$. P and $\sim P$ are a pair of predicates with fuzzy negative connection. [1,3]

1.2 An Infinite-Valued Model of Medium Predicate Logic (MF)

Definition 1. Let A be a formula of Medium predicate Logic. The λ -valuation \mathfrak{X}_λ of A ($\lambda \in (0,1)$) consist of following assignations:

¹ The project is supported by the National Natural Science Fund (60575038).

- (1) for each symbol of the individual constant in A, assigning an object in D;
- (2) for each symbol of the n-place function in A, assigning an mapping from D^n to D;
- (3) for each symbol of the n-place predicate in A, assigning an mapping from D^n to $[0,1]$, and

(3.1) $\mathfrak{R}_\lambda(A)$ takes only a numerical value in $[0, 1]$ at a time when A is an atomic formula;

(3.2) $\mathfrak{R}_\lambda(A) + \mathfrak{R}_\lambda(\neg A) = 1$;

(3.3) $\mathfrak{R}_\lambda(\sim A) =$

$$\frac{2\lambda - 1}{1 - \lambda} (\mathfrak{R}_\lambda(A) - \lambda) + 1 - \lambda, \quad \text{when } \lambda \in [\frac{1}{2}, 1), \text{ and } \mathfrak{R}_\lambda(A) \in (\lambda, 1];$$

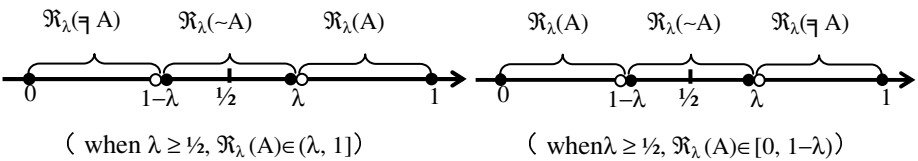
$$\frac{2\lambda - 1}{1 - \lambda} \mathfrak{R}_\lambda(A) + 1 - \lambda, \quad \text{when } \lambda \in [\frac{1}{2}, 1) \text{ and } \mathfrak{R}_\lambda(A) \in [0, 1 - \lambda];$$

$$\frac{1 - 2\lambda}{\lambda} \mathfrak{R}_\lambda(A) + \lambda, \quad \text{when } \lambda \in (0, \frac{1}{2}] \text{ and } \mathfrak{R}_\lambda(A) \in [0, \lambda];$$

$$\frac{1 - 2\lambda}{\lambda} (\mathfrak{R}_\lambda(A) + \lambda - 1) + \lambda, \quad \text{when } \lambda \in (0, \frac{1}{2}] \text{ and } \mathfrak{R}_\lambda(A) \in (1 - \lambda, 1];$$

(3.4) $\mathfrak{R}_\lambda(A \rightarrow B) = \text{Max}(1 - \mathfrak{R}_\lambda(A), \mathfrak{R}_\lambda(B))$, where B is a formula of MF;

We can realize the meaning of the model clearly by pictures as follows (suppose $\lambda \geq \frac{1}{2}$)



2 Medium Logic Representation of Fuzzy Knowledge

2.1 Expression Rule

Each kind of fuzzy knowledge in real life can be represented by fuzzy predicate in medium predicate logic. Based on the conjunctions “ \neg ”, “ \wedge ”, “ \vee ”, “ \rightarrow ”, “ \leftrightarrow ” and so on in traditional logic, it introduces fuzzy privative “ \sim ”, opposition privative “ \neg ” so as to extend the traditional logical expression and fetch more complicated well formed formula to express more complicated knowledge and concepts. Based on medium logic, “ \neg ” can be defined by “ \sim ” and “ \neg ”, viz. $\neg A =_{\text{df}} A \rightarrow \sim A$ or $\neg A =_{\text{df}} \eta A \vee \sim A$. According to its own domain, the paper will endow each fuzzy predicate expression with a truth value which belongs to interval $[0, 1]$ so as to realize application of the extended well formed formulas.

Definition 2. (1) Introduce symbol “ \sim^+ ”, for each fuzzy predicate A, \sim^+A denotes the part close to A in the medium $\sim A$.

(2) Introduce symbol “ \sim^- ”, for each fuzzy predicate A, \sim^-A denotes the part close to ηA in the medium $\sim A$.

Definition 3. In the infinite-valued model of Medium predicate Logic (MF), A is fuzzy predicate, for each x that belongs to discourse domain, it prescribes that:

(1) The truth value set of $\sim^+A(x)$ or \sim^-A is $\{0, 1, 1/2\}$. Suppose $\mathfrak{R}_\lambda(A(x))$ denotes the truth value of $A(x)$, if $1/2 < \lambda < 1$, so:

$$\mathfrak{R}_\lambda(\sim^+A(x)) = \begin{cases} 0 & \text{when } 1-\lambda < \mathfrak{R}_\lambda(A(x)) < 1/2 \\ 1/2 & \text{when } \mathfrak{R}_\lambda(A(x)) = 1/2 \\ 1 & \text{when } 1/2 < \mathfrak{R}_\lambda(A(x)) < \lambda \end{cases}$$

$$\mathfrak{R}_\lambda(\sim^-A(x)) = \begin{cases} 0 & \text{when } 1/2 < \mathfrak{R}_\lambda(A(x)) < \lambda \\ 1/2 & \text{when } \mathfrak{R}_\lambda(A(x)) = 1/2 \\ 1 & \text{when } 1-\lambda < \mathfrak{R}_\lambda(A(x)) < 1/2 \end{cases}$$

(2) $\mathfrak{R}_\lambda(\sim A(x)) = 1$ iff $1-\lambda < \mathfrak{R}_\lambda(A(x)) < \lambda$.

Remark: definition 2 and definition 3 are proposed to divide up medium information in further way. In order to compute easily, we give definition 3. In fact, we can also give a many-valued even infinite-valued function correspond to special demand.

2.2 Application

For introducing the medium logic representation of fuzzy knowledge more conveniently and embodying its effect, we take a simple decision-making program of financial investment for example, which represents many problems in real life.

The function of this program is to assist consumer to decide if to bank the money or to invest the money on stock market. Some investors may use their superfluous money in the two ways. Suppose the strategies of investors lie on their income and actual bank savings, which also follow the following rules:

- (1) If investors have a little savings, they should bank the superfluous money how much ever their income is.
 - (2) Those who have much savings and much income can consider investing their superfluous money on stock market that takes a risk but is very profitable.
 - (3) Those who have a little more savings and whose income is not low may consider using their most superfluous money to buy stocks and banking rest money.
 - (4) Those who have a little less savings and whose income is also not low may consider banking their most superfluous money and using rest money to buy stocks.
- (If investors have a little income, we think that they have no superfluous money.)

Medium logic representation. For realizing automatization of the decision-making program, we represent these guidelines by predicate calculus propositions. For differentiating symbols of predicates from symbols of functions more conveniently, symbols of predicates are written by capital letters while symbols of functions are written by small letters.

Individuals are seen as discourse domain, for each X that belongs to discourse domain, introduce necessary functions and predicates. Thereinto, $mfsavings(X)$ denotes money banked by X ; $mfstocks(X)$ denotes money used by X to buy stocks;

$MORE(mfsavings(X),mfstocks(X))$ denotes that money banked by X is more than that used to buy stocks.

Based on definition 2, $\sim^+MUCH(\text{savings}(X))$ denotes that X has a little more savings; $\sim^-MUCH(\text{savings}(X))$ denotes that X has a little less savings. And the same to $MUCH(\text{income}(X))$.

Those four strategies of investment can be expressed by logic expression with these predicates and functions. Rule (1) can be expressed as:

$$\neg MUCH(\text{savings}(X)) \rightarrow INVESTMENT(X, \text{savings}).$$

Also, rule (2) can be expressed as:

$$MUCH(\text{savings}(X)) \wedge MUCH(\text{income}(X)) \rightarrow INVESTMENT(X, \text{stocks}).$$

Rule(3):

$$\sim^+MUCH(\text{savings}(X)) \wedge (MUCH(\text{income}(X)) \vee \sim^-MUCH(\text{income}(X))) \rightarrow (INVESTMENT(X, \text{stocks}) \wedge INVESTMENT(X, \text{savings}) \wedge MORE(\text{mfsocks}(X), \text{mfsavings}(X))).$$

Rule(4):

$$\sim^-MUCH(\text{savings}(X)) \wedge (MUCH(\text{income}(X)) \vee \sim^+MUCH(\text{income}(X))) \rightarrow (INVESTMENT(X, \text{stocks}) \wedge INVESTMENT(X, \text{savings}) \wedge MORE(\text{mfsavings}(X), \text{mfstocks}(X)))$$

Measure of truth scale. Obviously, predicate “*MUCH*” is fuzzy predicate. We randomly investigate people’s viewpoints on individual income and family savings and get result as table1.

We list 20 people’s investigation data. Integrate investigation data in the same province or municipality directly under the Central Government. And then, we can get data as table2.

Remark: Because error cannot be avoided absolutely, we ascertain a flexible interval for each kind of data which is marked respectively under the corresponding title in table2. It is believable that the more people we investigate, the more veracious integrated data are and the smaller flexible intervals are.

When measuring truth value of some concrete datum “x” for fuzzy predicate, *MUCH* or $\neg MUCH$, the paper adopts the concept of distance. In this way, naturally, for *MUCH*, the truer x is, the farther x is from data range for $\neg MUCH$.

It can also be found out from data in table2 that if an income is high in Shanghai, it must be high in other area. If an income is low in Anhui, it must be low in other area. So is savings. For the sake of this characteristic of the data, we adopt Euclidean distance of one dimension, which is expressed as $d(x, y)$, viz. $d(x, y) = |x - y|$.

For each investigation datum “x”, define $h_T(MUCH(x))$ as a function of truth value of “x” for fuzzy predicate *MUCH*. And,

$$h_T(MUCH(x)) = \begin{cases} 0 & x \leq \alpha_F + \varepsilon_F \\ \frac{d(x, \alpha_F + \varepsilon_F)}{d(\alpha_F + \varepsilon_F, \alpha_T - \varepsilon_T)} & \alpha_F + \varepsilon_F < x < \alpha_T - \varepsilon_T \\ 1 & x \geq \alpha_T - \varepsilon_T \end{cases}$$

In the same way, function of truth value of “x” for $\neg MUCH$ can be given.

$$h_F(MUCH(x)) = \begin{cases} 0 & x \geq \alpha_T - \varepsilon_T \\ \frac{d(x, \alpha_T - \varepsilon_T)}{d(\alpha_F + \varepsilon_F, \alpha_T - \varepsilon_T)} & \alpha_F + \varepsilon_F < x < \alpha_T - \varepsilon_T \\ 1 & x \leq \alpha_F + \varepsilon_F \end{cases}$$

Remark: α_T is the integrated datum regarded as the truest one for fuzzy predicate *MUCH* in table 2, and ε_T is its flexible extent. α_F is the integrated datum regarded as the truest one for fuzzy predicate \neg *MUCH*, and ε_F is its flexible extent.

Table 1.

People	Area	Viewpoint of much income (yuan/month)	Viewpoint of a little income (yuan/month)	Viewpoint of much savings (ten thousand yuan)	Viewpoint of a little savings (ten thousand yuan)
1	Nanjing, Jiangsu	≥ 10000	≤ 1500	≥ 20	≤ 8
2	Nanjing, Jiangsu	≥ 8000	≤ 1000	≥ 15	≤ 5
3	Wuxi, Jiangsu	≥ 12000	≤ 1200	≥ 15	≤ 8
4	Wuxi, Jiangsu	≥ 10000	≤ 1500	≥ 15	≤ 10
5	Suzhou, Jiangsu	≥ 15000	≤ 1500	≥ 15	≤ 10
6	Shanghai City	≥ 15000	≤ 2000	≥ 25	≤ 15
7	Shanghai City	≥ 15000	≤ 2500	≥ 20	≤ 10
8	Baoshan, Shanghai	≥ 10000	≤ 2000	≥ 20	≤ 8
9	Shanghai City	≥ 20000	≤ 2000	≥ 25	≤ 10
10	Jinshan, Shanghai	≥ 12000	≤ 1500	≥ 15	≤ 7
11	Chuzhou, Anhui	≥ 4000	≤ 800	≥ 10	≤ 5
12	Fuyang, Anhui	≥ 5000	≤ 1000	≥ 10	≤ 5
13	Hefei, Anhui	≥ 6000	≤ 1000	≥ 10	≤ 8
14	Hefei, Anhui	≥ 5000	≤ 1000	≥ 10	≤ 5
15	Huangshan, Anhui	≥ 5000	≤ 800	≥ 10	≤ 5
16	Jinan, Shandong	≥ 6000	≤ 1200	≥ 15	≤ 8
17	Jinan, Shandong	≥ 8000	≤ 1000	≥ 10	≤ 8
18	Weihai, Shandong	≥ 10000	≤ 1500	≥ 15	≤ 8
19	Linyi, Shandong	≥ 5000	≤ 800	≥ 10	≤ 5
20	Yantai, Shandong	≥ 6000	≤ 1000	≥ 12	≤ 5

Table 2.

Area	Viewpoint of much income(yuan/month) (± 500)	Viewpoint of a little income(yuan/month) (± 100)	Viewpoint of much savings(ten thousand yuan) (± 2)	Viewpoint of a little savings (ten thousand yuan) (± 1)
Jiangsu	≥ 11000	≤ 1340	≥ 16	≤ 8.2
Shanghai	≥ 14400	≤ 2000	≥ 21	≤ 10
Anhui	≥ 5000	≤ 920	≥ 10	≤ 5.6
Shandong	≥ 7000	≤ 1100	≥ 12.4	≤ 6.8

Obviously, $h_T(MUCH(x)) + h_F(MUCH(x)) = 1$, so $h_T(MUCH(x)) + h_T(\neg MUCH(x)) = 1$, which corresponds to the infinite-valued model of Medium predicate Logic (MF).

For the viewpoint of much income, the corresponding integrated datum of Shanghai is the highest, which is 14400. We regard it as the truest datum for *MUCH*, and its flexible extent ϵ_T is 500. For the viewpoint of a little income, the corresponding integrated datum of Anhui is the lowest, which is 920. We regard it as the truest datum for $\neg MUCH$ or the falsest datum for *MUCH*, and its flexible extent ϵ_F is 100.

So, for income,

$$h_T(MUCH(x)) = \begin{cases} 0 & x \leq 1020 \\ \frac{d(x, 1020)}{d(1020, 13900)} & 1020 < x < 13900 \\ 1 & x \geq 13900 \end{cases}$$

$$h_F(MUCH(x)) = h_T(\neg MUCH(x)) = 1 - h_T(MUCH(x)).$$

In the same way, for savings,

$$h_T(MUCH(x)) = \begin{cases} 0 & x \leq 6.6 \\ \frac{d(x, 6.6)}{d(6.6, 19)} & 6.6 < x < 19 \\ 1 & x \geq 19 \end{cases}$$

$$h_F(MUCH(x)) = h_T(\neg MUCH(x)) = 1 - h_T(MUCH(x))$$

And then, each investigation datum can get a truth value for corresponding predicate expression according to corresponding function of truth value.

Ascertainment of the “ λ ” in infinite-valued model of MF and its meaning. Take Jiangsu for example, viewpoint of much income is ≥ 11000 , and that of a little income is ≤ 1340 . According to function of truth value corresponding to income, we can educe (round off result to three decimals):

$$h_T(MUCH(11000)) = \frac{d(11000, 1020)}{d(1020, 13900)} = 0.775,$$

$$h_T(\neg MUCH(11000)) = h_F(MUCH(11000)) = 1 - 0.775 = 0.225;$$

$$h_T(MUCH(1340)) = \frac{d(1340,1020)}{d(1020,13900)} = 0.025;$$

$$h_T(\neg MUCH(1340)) = h_F(MUCH(1340)) = 1 - 0.975 = 0.975.$$

11000 and 1340 are a pair of datum of opposite concepts, so in theory, $h_T(MUCH(11000)) + h_T(MUCH(1340)) = 1$. Because investigation data are not enough, data themselves are not veracious enough, and model itself could not absolutely correspond with fuzzy fact, so some error is avoidless.

Prescribe that $\lambda = (0.775 + 0.975) / 2 = 0.875$, which represents the truth value of datum corresponding with viewpoint of much income in Jiangsu for fuzzy predicate *MUCH*. For each datum *x*, if $h_T(MUCH(x)) > 0.875$ or $h_T(\neg MUCH(x)) < 1 - 0.875 = 0.125$, it means that datum *x* belongs to high income in Jiangsu; if $h_T(MUCH(x)) < 0.125$ or $h_T(\neg MUCH(x)) > 0.875$, it means that *x* belongs to low income in Jiangsu.

So the meaning of ascertainment for “ λ ” in this case has been found, viz. area variety can be replaced by value variety of λ . For example, if an investigation datum is from Jiangsu, endow λ with corresponding value; if from Shandong, endow λ with corresponding value.

Work out values of these areas for viewpoint of income respectively. Result as Table 3.

Table 3.

Area (viewpoint of income)	Jiangsu	Shanghai	Anhui	Shandong
λ	0.875	0.962	0.655	0.729

For viewpoints of savings in these areas, values of λ can also be worked out respectively in the same way. Result as Table 4.

Table 4.

Area (viewpoint of savings)	Jiangsu	Shanghai	Anhui	Shandong
λ	0.815	0.863	0.637	0.726

3 Realization by Fuzzy Production Rules

For realization, we can consider to convert logic formulas into production rules. Obviously, when confidence level of rule is larger than value of λ , the value of λ can be regarded as threshold.

Example: One people lives in Wuxi, Jiangsu. His income is 5000 yuan a month and his family savings is 120 thousand. How does he invest?(Here suppose confidence levels of rule(1)-rule(4) are all 0.9 for convenience.)

First, the people lives in Jiangsu, so ascertain the value of λ_1 , viz. $\lambda_1 = 0.875$, when considering about income and the value of λ_2 , viz. $\lambda_2 = 0.815$, when considering about savings. Obviously, both λ_1 and λ_2 are smaller than confidence level of rules.

According to corresponding function of truth value and infinite-valued model of MF, educe results as follows:

$$h_T(MUCH(savings(X))) = \frac{d(12, 6.6)}{d(6.6, 19)} = 0.435; h_T(\neg MUCH(savings(X))) = 0.565;$$

$$h_T(MUCH(income(X))) = \frac{d(5000, 1020)}{d(1020, 13900)} = 0.309; h_T(\neg MUCH(income(X))) = 0.691;$$

For rule (1):

$$INVESTMENT(X, savings) \leftarrow \neg MUCH(savings(X)), \quad 0.9, \quad 0.815;$$

Prescribe that $t = \min\{0.565, 0.9\} = 0.565 < 0.815$, so the rule can not act.

Rule (2):

$$INVESTMENT(X, stocks) \leftarrow MUCH(savings(X)) \wedge MUCH(income(X)), \quad 0.9, \quad 0.875;$$

Remark: Here premises come down to savings and income, so threshold

$$\tau = \max\{0.875, 0.815\} = 0.875.$$

Truth value of premises

$$t_1 = \min\{h_T(MUCH(savings(X))), h_T(MUCH(income(X)))\} = \min\{0.435, 0.309\} = 0.309,$$

$$t = \min\{0.309, 0.9\} = 0.309 < 0.875,$$

So the rule can not act.

Rule (3):

INVEST-

$$MENT(X, stocks) \wedge INVESTMENT(X, savings) \wedge MORE(mfstocks(X), mfsavings(X)) \leftarrow \sim^+ MUCH(savings(X)) \wedge (MUCH(income(X)) \vee \sim MUCH(income(X))), \quad 0.9, \quad 0.875;$$

Because $1 - 0.815 < h_T(MUCH(savings(X))) < 0.5$, according to definition 3, $h_T(\sim^+ MUCH(savings(X))) = 0$. Obviously, the rule can not act.

Rule (4):

$$INVESTMENT(X, stocks) \wedge INVESTMENT(X, savings) \wedge MORE(mfsavings(X), mfstocks(X)) \leftarrow \sim MUCH(savings(X)) \wedge (MUCH(income(X)) \vee \sim MUCH(income(X))), \quad 0.9, \quad 0.875;$$

Because $1 - 0.815 < h_T(MUCH(savings(X))) < 0.5$, according to definition 3, $h_T(\sim MUCH(savings(X))) = 1$, and $1 - 0.875 < h_T(MUCH(income(X))) < 0.875$, so $h_T(\sim MUCH(income(X))) = 1$. So truth value of premises is 1. For $t = 0.9 > 0.875$, the rule can act.

In conclusion, this people can adopt rule (4) for financing.

References

1. Zhu, W.J., Xiao, X.A.: Propositional Calculus System of Medium Logic (I). Nature Magazine 8, 315–316 (1985)
2. Zadeh, L.: Fuzzy Sets. Information and Control, 338–353 (1965)
3. Zhu, W.J., Xiao, X.A.: Essentials of Mathematical Foundation. Nanjing University Press, Nanjing (1996)

4. Zhu, W.J., Xiao, X.A.: An Interpretation of Infinite Values for Medium Proposition Logic. In: Proc. of IEEE-Third International Conference on Machine Learning and Cybernetics, vol. 4(7), pp. 2495–2499 (2004)
5. Long, H., Xian, X., Zhu, W.J.: Measure of Medium Truth Scale and Its Application(I). Chinese Journal of Computers, 2186–2193 (2006)
6. He, X.G.: Fuzzy Theories and Fuzzy Techniques in Knowledge Processing, 2nd edn. National defense industries Press, Beijing (1998)
7. Chen, S.F., Chen, Z.Q.: Artificial Intelligence and Knowledge Engineering. Nanjing University Press, Nanjing (1997)

Five Perspectives on Case Based Reasoning

Zhaohao Sun¹, Jun Han², and Dong Dong¹

¹ School of Computer Science and Technology, College of Mathematics and Information Science, Hebei Normal University, Shijiazhuang, China, 050016
zhsun@ieee.org, dongdong812@126.com

² School of Computer Science, Beihang University, Beijing 100083, China
jun_han@buaa.edu.cn

Abstract. This article examines case-based reasoning (CBR) from five different perspectives: cognitive, process-oriented, logical, intelligent, and hybrid perspective. These five perspectives cover the majority of CBR research and development and classify CBR into five categories consequently: Cognitive CBR, process CBR, logical CBR, intelligent CBR and hybrid CBR. This article looks at each of these perspectives on CBR from a unified viewpoint taking into account corresponding fundamentals, applications and related issues. The proposed approach can facilitate research and development of CBR and its applications.

Keywords: Case-based reasoning (CBR), fuzzy logic, artificial intelligence, multiagent system, soft computing.

1 Introduction

Case based reasoning (CBR) has drawn considerable attention in artificial intelligence (AI) with many successful applications in customer support [1][3][12], online sales [2], e-commerce [24] and multiagent systems [21]. However, many of these applications are still at an empirical level; that is, they try to provide examples of using CBR terminology to explain the real world scenarios such as in e-commerce applications. Further, different researchers usually examine one real world problem from different perspectives, which is also true in CBR. At least, CBR has been studied from the following different perspectives: cognitive perspective [10][11][22], process-oriented perspective [1][26], logical perspective [6][25] and intelligent perspective [24][25], although the researchers might have not been aware of them, and there has not been a unified treatment for CBR to look at these perspectives.

This article will fill the above-mentioned gap by providing a unified treatment for these perspectives. Each of them corresponds to special form of CBR; that is, cognitive CBR, process CBR, logical CBR, intelligent CBR and hybrid CBR, which will be used respectively in the context of the article. The proposed approach in this article can facilitate research and development of CBR and its applications.

The rest of the article is organized as follows: Section 2 to 6 examines cognitive CBR, process CBR, logical CBR, intelligent CBR and hybrid CBR respectively. Section 7 discusses the interrelationship among five perspectives on CBR. Section 8 concludes this paper with some concluding remarks and future work.

2 Cognitive CBR

This section examines CBR from a cognitive perspective, we refer to it as cognitive CBR, which is a philosophical basis for all other perspectives on CBR. Cognitive CBR is also the beginning of CBR, and still affects the research and development of CBR.

CBR arose out of the research of cognitive science [19] (p. 1). Cognitive CBR was studied by Roger Shank's group at Yale University at the early 1980s [29] (p. 18). At that time, Shank, et al. proposed a cognitive model for CBR: Dynamic memory model, which suggests that events or cases are understood in terms of scripts and other knowledge structures as well as relevant previous experiences [22]. Then Kolodner developed the first cognitive CBR systems, CYRUS, based on the above-mentioned cognitive model [10]. CYRUS contains cases of some travels or meetings and allows users to ask questions about these events [29]. Shank, et al. demonstrate the basis for cognitive CBR that when facing with a new problem, people merely search their memories for past problems resembling the current problem, recalling past experiences that somehow remind them of the current situations, which is similar to what Plato thought two thousand years ago [24] (p. 13), and revise or adapt the prior solution to fit the current problem [3] (p. 328). Therefore, the principle behind the above mentioned problem solving strategy is: Similar problems have similar solutions [26]. This can also be considered as the basis of a cognitive CBR [10].

In the 1980s, the US DARPA funded a series of workshops on CBR and the development of a CBR tool, ReMind, which can be considered as an important result of cognitive CBR, since ReMind was joined in the marketplace almost immediately by several other tools. The foundation of cognitive CBR has been summarized in Kolodner's work [10] in 1994. However, how to formalize CBR from a logical and intelligent perspective was still an open problem at that time. With the development of expert systems and intelligent systems [16], CBR has realized its shift from cognitive CBR to other forms of CBR.

3 Process CBR

CBR came from research in cognitive science and knowledge representation in the USA in the 1970s, and cognitive CBR was essentially developed in the 1980s in the USA. The European scientists try to understand CBR from a process perspective in the 1990s, which forms process CBR.

In the history of CBR, the CBR researchers have developed the CBR systems based on methodology of rule-based expert systems (RBES) and software engineering, both of the latter have been drawn important attention in the 1980s and 1990s. Furthermore, Aamodt, et al. summarize CBR from a process perspective and propose a R^4 -CBR cycle [1]. Finnie and Sun [8] extends the R^4 -CBR cycle to the R^5 -CBR cycle by adding case repartition, because building and maintaining a case base is a big issue in the situation where there are mass cases, for instance, the mass data/knowledge on the Web. Case repartition can reduce the complexity of case base and facilitate the case retrieval. This R^5 -CBR cycle consists of (case) repartition,

retrieve, reuse, revise and retain, as shown in Fig. 1. [8]. Based on this model, a typical CBR system at least consists of the following five R process stages:

1. Repartition a case base to form a satisfactory one
2. Retrieve the case base for the most similar case
3. Reuse the cases to attempt to solve the current problem
4. Revise the proposed solution
5. Retain the new solution as a part of a new case.

It should be noted that retrieve, revise, reuse and retain are the basic operations for managing case base (CB) and, however, not the unique feature of CBR systems, because they are also used in database management systems [27].

So far, we have not found the feature of “reasoning” in CBR: Where can one find logical reasoning such as modus ponens in the CBR systems? In other words, CBR is a paradigm without any logical reasoning in many publications. Therefore, CBR should be examined from a logical perspective.

4 Logical CBR

Case-based reasoning (CBR) is, first of all, a kind of reasoning from its name. Reasoning is an important problem solving tool in mathematics and logic. Reasoning also is an important tool in AI, in which it is mainly based on the reasoning in mathematical logic [16]. Based on this idea, CBR has a close relation with mathematical logic, which is the basis for applying fuzzy logic to CBR [19][24]. Therefore, it is necessary to examine CBR from a logical perspective. The corresponding CBR is called logical CBR, which used to be basically neglected in the CBR community. For example, one cannot find any logical model in some CBR related books. However, in the past few years, Sun and Finnie provide the foundations for logical CBR [9][24].

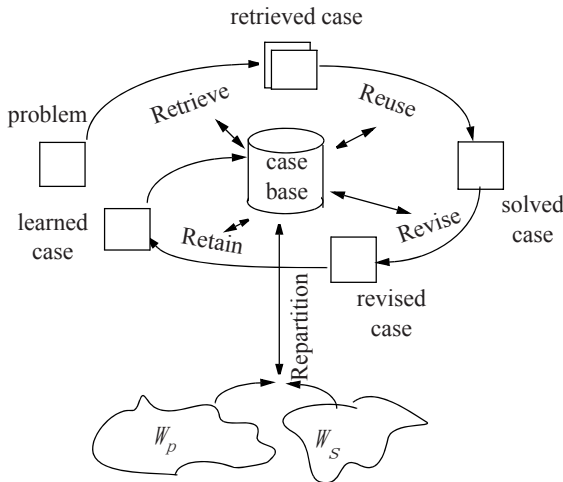


Fig. 1. The R^5 model of CBR cycle [8]

CBR is a reasoning paradigm based on previous experiences or cases; namely, a CBR system solves new problems by adapting solutions that were used to solve old problems [6][10]. Therefore, we call CBR a form of experience-based reasoning (EBR) [23]:

$$\text{CBR} := \text{Experience-based reasoning} \quad (1)$$

In business activities, it is usually true that “Two cars with similar quality features have similar prices.” This is a kind of EBR, essentially, a kind of similarity-based reasoning (SBR). In other words, SBR can be considered as a special and operational form of EBR. Therefore, CBR can be considered as a kind of similarity-based reasoning from a logical viewpoint [9].

$$\text{CBR} := \text{Similarity-based reasoning} \quad (2)$$

One of the most important inference rules for deduction is modus ponens [16]:

$$\frac{P \rightarrow Q}{P} \quad \therefore Q \quad (3)$$

The inference rule for SBR can be considered as generalized modus ponens [24]:

$$\frac{P \rightarrow Q}{P} \quad \therefore Q \quad (4)$$

where P , P' , Q and Q' represent compound propositions, Q and Q' are similar in the sense of a certain similarity [7]. This is a theoretical foundation for CBR, in particular for similarity-based case retrieval.

When P , P' , Q and Q' represent fuzzy propositions, Q and Q' are similar in the context of fuzzy logic, then Eq. (4) is also a theoretical foundation for fuzzy CBR or soft CBR [19].

Although there are many researches on logical CBR, there is still no answer to what is first order case-based logic. This is an interesting topic but yet an open problem, because any reasoning paradigm should find its original form in logic. Hence, CBR as a reasoning paradigm should be considered as a part of case-based logic.

It should be noted that some researchers confuse CBR with its corresponding system. In fact, a CBR system is an implementation of applying engineering principles to logical CBR or a software implementation of logical CBR based on intelligent techniques. Briefly, similar to the inference engine (IE) and knowledge base (KB) in expert systems (ES), one can also use CBR engine (CBRE) and a case base (CB) to denote a CBR system (CBRS); that is,

$$\text{CBRS} = \text{CB} + \text{CBRE} \quad (5)$$

where the CBRE performs similarity-based reasoning, whereas the IE in traditional ESs mainly performs traditional deductive reasoning [24].

5 Intelligent CBR

CBRSs are intelligent systems (ISs) because they are a special kind of ESs that are ISs [24]. Intelligent CBR systems (ICBRS) aim to address some of the drawback of rule based systems (RBS) [3] (p. 161)[26]. Therefore, ICBRS is an improvement of RBS, which are a part of ESs. This implies that ICBRS is an extended form of ESs. Therefore, the interrelationship among ISs, ESs and ICBRS can be shown as in Fig. 2.

From the viewpoint of ES, an IS consists of two parts: One is a KB, the other is an IE. The general architecture of an ICBRS based on the process model of CBR is as shown in Fig. 3. This architecture extends the knowledge-based model of CBR [24](p. 252). In this architecture, similar to database manager [27], the CB manager in the intelligent CBR system realizes the R^5 - CBR cycle in Section 3. Similar to the IE in ES [16], the CBR engine is the reasoning mechanism for manipulating cases in the CB based on the SBR mentioned in Section 4. Therefore, ICBRSs integrate database systems and knowledge based systems (KBSs) or ESs.

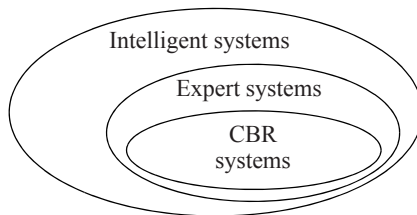


Fig. 2. Interrelationships between CBRS, ES and IS

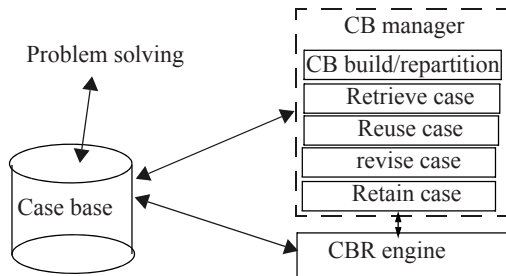


Fig. 3. A general architecture of intelligent CBR system

It should be noted that many intelligent CBR systems lack either CBR engine or CB manager. For example, WEBSSELL is an intelligent CBR sales assistant for the WWW [2]. However, it confuses CB manager with CBR engine. The confusion of CBR manager and CBR engine hinders the interrelationship of CBR systems with database systems and KBSs.

6 Hybrid CBR

Hybrid CBR has drawn attention since the end of last century. Since then, there are combinations of different reasoning methods, including database technology, rules, analogical reasoning, deep causal reasoning, genetic algorithms and neural nets, with CBR [31]. All these combinations can be considered as hybrid CBR. Generally, any application of CBR in practice such as e-commerce, customer support, e-services requires a hybrid form of CBR, because it is difficult for any applied system to use a single reasoning paradigm. Furthermore, process CBR and intelligent CBR can be considered as a hybrid form of logical CBR, because the former is a combination of logical CBR and process technique, whereas the latter integrates logical CBR and knowledge-based system (KBS). Thus, it is of significance to examine CBR from a hybrid perspective, which corresponds to hybrid CBR. However, owing to the space limitation, we can only briefly examine some main hybrid CBR paradigms; that is, rule-based CBR, soft CBR, multiagent CBR and CBR as process reasoning.

6.1 Rule Based CBR

Rule-based reasoning (RBR) is one of the oldest reasoning paradigms for knowledge processing and KBSs in AI [16]. Rule-based CBR combining RBR with CBR has drawn attention since the early 1980s. From a theoretical viewpoint, CBR is an extended form of RBR taking into account SBR (Section 4). However, the combination of RBR and CBR is still drawn attention in the CBR community. For example, Luo, et al. [13] combine CBR and RBR to construct a quick emergency response plan generation system, in which, the CBR subsystem is used to generate decision support from the previous emergency cases and solutions in database through the similarity retrieval, while the RBR subsystem is used to reason for solution and then the reasoning result is stored in the case base for future reuse.

6.2 Soft CBR

Fuzzy reasoning has drawn a lasting attention to AI since the 1970's, and found countless applications in various fields such as family electronic appliance, decision making, ESs and databases [30]. In order to move CBR towards a theoretical foundation, Dubois, et al. propose fuzzy set-based models for CBR and a logical formalization of the basic CBR inference [6]. Plaza, et al. propose a logical approach to CBR using fuzzy similarity relations [20]. However, these studies have no intention of building the logical correspondence between the mentioned models, although they are based on fuzzy logic or SBR. The core of fuzzy reasoning is the generalized form of modus ponens taking into account fuzzy knowledge existed in AI. Therefore, the model of fuzzy reasoning is similar to that of SBR (Eq. (4)) although they have had different semantics. The further development of fuzzy reasoning or fuzzy logic is soft computing. How to combine soft computing with CBR becomes an interesting topic in the end of 1990s. Pal and Shiu provide the foundation of applying soft computing to CBR [19]. However, their attempts are not based on logical foundations of CBR. It seems that relationships between logical CBR, fuzzy CBR and soft CBR are still a big issue in this area.

6.3 CBR as a Process Reasoning

CBR can be considered as an integration of process technology and SBR paradigms. However, CBR is not only a SBR, but it is a kind of process reasoning [8], which is a more complex reasoning paradigm that is usually used in intelligent systems.

A process reasoning is a kind of reasoning that infers information about a domain using process or multistage methods and there exists a reasoning paradigm that plays a vital role in every main stage of the process [9].

A typical reasoning in CBR, also named as the R^s-CBR cycle (Section 3), mainly consists of (case) Repartition, Retrieve, Reuse, Revise and Retain. Each of these five components is a complex process. For example, case retrieval is a complex operation in the case base. Furthermore, case retrieval and case adaptation are two main stages in CBR [4], in which SBR plays an important role. For instance, case retrieval is based on SBR [5][7], case adaptation is also based on SBR, but maybe on a different SBR [25]. In fact, case base repartition and building [18] is also based on SBR [24]. Thus, CBR is a process reasoning, in which SBR dominates each of main stages: case base repartition and building, case retrieval, and case adaptation.

6.4 Multiagent CBR

Multiagent system (MAS) is a moving AI technique since the end of the 1990s [24][28]. Combining CBR with MAS has drawn attention to the CBR community. For example, Matos and Sierra propose a case-based architecture to model agent negotiation in MAS [15]. Plaza, et al. examine the collaborative policies for multiagent cooperative CBR [21]. Generally, multiagent CBR is still at the early stage, because there are still many big issues to be solved such as cooperation, negotiation and brokerage among the CBR agents and other kinds of agents in MAS. However, with the dramatic development of the Internet/Web, multiagent CBR systems will play an important role in online retrieval, case based negotiation and online recommendation [24].

It should be noted that the above-mentioned forms of hybrid CBR can be considered as applied CBR. In fact, any application system based on CBR is an integration of CBR with some intelligent techniques. Further, multiagent CBR can be considered as a form of intelligent CBR, because MAS is a part of AI to some extent.

7 Interrelationships among Five Perspectives on CBR

Corresponding to the five perspectives on CBR we have examined cognitive CBR, process CBR, logical CBR, intelligent CBR and hybrid CBR. Cognitive CBR can be considered as the beginning of CBR. Logical CBR is a realization of cognitive CBR based on mathematical logic and provides a theoretical foundation of CBR. Process CBR provides application potentials for CBR. Intelligent CBR transforms CBR from theory to practice based on logical CBR, process CBR and intelligent techniques to solve some intelligent problems in the real world. Hybrid CBR promotes the combination of CBR with other intelligent techniques. These interrelationships among them can be illustrated in Fig. 4.

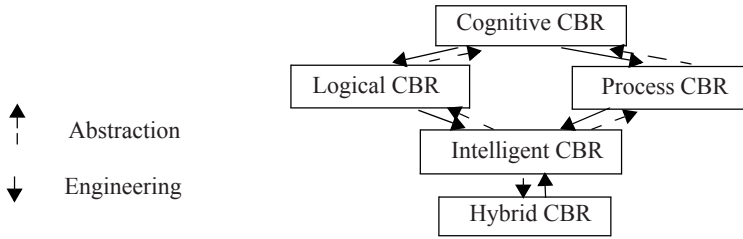


Fig. 4. A unified model for five perspectives on CBR

There are two different arrows in Fig. 4. The dashed arrow represents the abstraction or generalization process. The solid arrow denotes engineering process. More specifically, cognitive CBR is transformed into hybrid CBR system through the engineering process based on logical CBR, process CBR and intelligent CBR. Conversely, hybrid CBR moves to cognitive CBR through abstraction or generalization process based on intelligent CBR, logical CBR and process CBR.

It should be noted that any transformation from one form of CBR to another requires either techniques of computer science (for top-down transformation) or techniques of scientific discovery (for bottom-up transformation).

8 Concluding Remarks

This article examined five perspectives on CBR from a unified viewpoint. These five perspectives classify CBR into five categories: Cognitive CBR, process CBR, logical CBR, intelligent CBR and hybrid CBR. This proposed approach and classification can facilitate the research and development of CBR with applications in e-commerce, customer support and web services recommendation. In the future work, we will further examine interrelationships among these five perspectives on CBR taking into account corresponding theory, practice, tools and related issues. We will also examine hybrid CBR in more detail.

References

- [1] Aamodt, A., Plaza, E.: Case-based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *Artif. Intell. Comm.* 7(1), 39–59 (1994)
- [2] Cunningham, P., Bergmann, R., Schmitt, S., Breen, S., Smyth, B.: *Traphoener: Intelligent Support for Online Sales: The Websell Experience* (2001), <http://www.aic.nrl.navy.mil/papers/2001/AIC-01-003/ws3/ws3toc6.pdf>
- [3] Bergmann, R., Schmitt, S., Stahl, A.: Intelligent Customer Support for Product Selection with Case-based Reasoning. In: *E-commerce and Intelligent Methods*, pp. 322–341. Physica-Verlag, Heidelberg (2002)
- [4] Craw, S., Wiratuga, N., Rowe, R.C.: Learning Adaptation Knowledge to Improve Case-based Reasoning. *J. Artif. Intell.* 170, 1175–1192 (2006)
- [5] Dubois, D., Prade, H.: *Similarity-Based Approximate Reasoning*. Computational Intelligence: Imitating life, pp. 69–80. IEEE Press, New York (1994)

- [6] Dubois, D., Esteva, F., Garcia, P., Godo, L., López de Màntaras, R., Prade, H.: Case-based Reasoning: A Fuzzy Approach. In: Ralescu, A.L., Shanahan, J.G. (eds.) *Fuzzy Logic in Artificial Intelligence. IJCAI 1997 Workshop*, pp. 69–80. Springer, Berlin (1999)
- [7] Finnie, G., Sun, Z.: Similarity and Metrics in Case-based Reasoning. *Int. J. Intell. Syst.* 17(3), 273–287 (2002)
- [8] Finnie, G., Sun, Z.: R^S Model for Case-based Reasoning. *Knowledge-based Syst.* 16(1), 59–65 (2003)
- [9] Finnie, G., Sun, Z.: A Logical Foundation for the CBR Cycle. *Int. J. Intell. Syst.* 18(4), 367–382 (2003)
- [10] Kolodner, J.: *Case Based Reasoning*. Morgan Kaufmann, San Mateo (1993)
- [11] Leake, D.: *Case-Based Reasoning: Experiences, Lessons & Future Direction*. AAAI Press/MIT Press, Menlo Park (1996)
- [12] Lenz, M., Bartsch-Spörl, B., Burkhard, H.D., Wess, S. (eds.): *Case-Based Reasoning Technology, from Foundations to Applications*. Springer, Berlin (1998)
- [13] Luo, J., Shi, Z.P., He, Q., Shi, Z.Z.: A Quick Emergency Response Plan Generation System Combining CBR and RBR (in Chinese). *J. Computer R & D* 44(4), 660–666 (2007)
- [14] Novais, P., Brito, L., Neves, J.: Agreement in Virtual Marketplaces with CBR-supported Negotiation. In: *Proc. PAAM 2000- 5th Int. Conf. on the Practical Application of Intelligent Agents and Multiagents*, Manchester, pp. 203–206 (2007)
- [15] Matos, N., Sierra, C.: Evolutionary Computing and Negotiating Agents. In: Noriega, P., Sierra, C. (eds.) *AMET 1998 and AMEC 1998. LNCS (LNAI)*, vol. 1571, pp. 126–150. Springer, Heidelberg (1999)
- [16] Nilsson, N.J.: *Artificial Intelligence: A New Synthesis*. Morgan Kaufmann Publishers, San Francisco (1998)
- [17] Olivia, C., Chang, C.F., Enguis, C.F., Ghose, A.K.: Case-based BDI Agents: AN Effective Approach for Intelligent Search on the World Wide Web. In: *AAAI Spring Symposium on Intelligent Agents in Cyberspace*, CA, pp. 20–27. AAAI Press, Menlo Park (1999)
- [18] Pan, R., Yang, Q., Pan, S.J.: Mining Competent Case Bases for Case-based Reasoning. *Artif. Intell.* 171, 1039–1068 (2007)
- [19] Pal, S.K., Shiu, S.C.K.: *Foundations of Soft Case-Based Reasoning*. John Wiley & Sons, Hoboken (2004)
- [20] Plaza, E., Esteva, F., Garcia, P., Godo, L., López de Màntaras, R.: A Logical Approach to Case-based Reasoning Using Fuzzy Similarity Relations. *Info. Scie.* 106, 105–122 (1996)
- [21] Plaza, E., Ontañón, S.: Ensemble Case-based Reasoning: Collaboration Policies for Multiagent Cooperative CBR. In: Aha, D.W., Watson, I. (eds.) *ICCBR 2001. LNCS (LNAI)*, vol. 2080, pp. 437–451. Springer, Heidelberg (2001)
- [22] Shank, R.: *Dynamic Memory: A Theory of Learning in Computers and People*. Cambridge University Press, New York (1982)
- [23] Sun, Z., Finnie, G.: A fuzzy Logic Approach to Experience Based Reasoning. *Int. J. Intell. Syst.* 22(8), 867–889 (2007)
- [24] Sun, Z., Finnie, G.: *Intelligent Techniques in E-Commerce: A Case-based Reasoning Perspective*. Springer, Heidelberg (2004)
- [25] Sun, Z., Finnie, G.: A Unified Logical Model for CBR-based E-commerce Systems. *Int. J. Intell. Syst.* 20(1), 29–46 (2005)
- [26] Watson, I.: An Introduction to Case-based Reasoning. In: Watson, I. (ed.) *Progress in Casebased Reasoning*, pp. 3–16. Springer, Berlin (1995)

- [27] Ullman, J.D., Widom, J.: *A First Course in Database Systems*, 2nd edn. Prentice Hall, Englewood Cliffs (2002)
- [28] Weiss, G. (ed.): *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. The MIT Press, Cambridge (1999)
- [29] Wilke, W., Lenz, M., Wess, S.: Intelligent sales support with CBR. In: Lenz, M., Bartsch-Spörl, B., Burkhard, H.D., Wess, S. (eds.) *Case-based Reasoning: From Foundations to Applications*, pp. 91–113. Springer, Berlin (1998)
- [30] Zimmermann, H.J.: *Fuzzy set theory and its application*. Kluwer, Boston (1991)
- [31] <http://www.ai-cbr.org/biblio/Hybrid%20CBR.html> (accessed April 06, 2008)

Fixpoint Semantics and Completeness of the Computational Model for Fuzzy Linguistic Logic Programming

Van Hung Le¹, Fei Liu¹, and Dinh Khang Tran²

¹ Department of Computer Science and Computer Engineering
La Trobe University

Bundoora, VIC 3086, Australia

vh2le@students.latrobe.edu.au, f.liu@latrobe.edu.au

² Faculty of Information Technology

Hanoi University of Technology

Hanoi, Vietnam

khangtd@it-hut.edu.au

Abstract. Fuzzy linguistic logic programming, which is a result of integrating hedge algebras and fuzzy logic programming, is proposed to facilitate the representation and reasoning on knowledge expressed in natural language, in which vague sentences are usually given a degree of truth stated in linguistic terms rather than a number, and linguistic hedges are very often used. To compute the truth value of a query, a computational model which directly manipulates linguistic terms is provided. The computational model has been proved to be sound. This paper presents a fixpoint semantics for fuzzy linguistic logic programs and based on it proves the completeness of the computational model.

Keywords: fuzzy logic programming; hedge algebra; linguistic value; linguistic hedge; fixpoint semantics; completeness.

1 Introduction

People use words in natural language, which are inherently imprecise, vague and qualitative in nature, to describe real world information, to analyse, to reason, and to make decisions. Moreover, linguistic hedges such as *very*, *more or less*, *quite*, and *rather* are very often used to state different levels of emphasis. Therefore, it is necessary to investigate logical systems that can directly work with words and make use of linguistic hedges since such logical systems will make it easier to represent and reason on knowledge expressed in natural language.

In fuzzy linguistic logic programming (FLLP), introduced in [3], each fact or rule in a logic program is assigned a truth value which is a linguistic term such as *true*, *very true*, *more or less true*, and *false* taken from a hedge algebra (HA) [6,7] of the truth variable, and linguistic hedges, which are not allowed in fuzzy logic programming (FLP) [9], can be used as unary connectives in formulae. It is shown in [3] that a knowledge base expressed in natural language can be easily

represented by the language. In this paper, we present a fixpoint semantics for logic programs and prove the completeness of the computational model.

The paper is organised as follows. Section 2 discusses linguistic truth domains, connectives and recalls some fundamental notions of FLLP. Section 3 presents a fixpoint semantics of logic programs while Section 4 proves the completeness of the computational model. Section 5 briefly concludes the paper.

2 Preliminaries

2.1 Linguistic Truth Domains and Connectives

The logic is truth-functional, i.e., the truth value of a compound formula, built from its components using a logical connective, is a function, which is called the truth function of the connective, of the truth values of the components.

As presented in [3], a linguistic truth domain for FLLP is a finite and linearly ordered set $\overline{X} = X \cup \{0, W, 1\}$, where X is a terms-domain of a K -limited monotonic HA $(X, \{c^-, c^+\}, H, \leq)$ (note that $c^- = False, c^+ = True$), and 0, W , and 1 are the least, the neutral and the greatest elements of \overline{X} , respectively.

The usual *conjunction* and *disjunction* are defined as $x \wedge y = \min(x, y)$ and $x \vee y = \max(x, y)$. The truth function of a hedge connective h is its inverse mapping $h^- : \overline{X} \rightarrow \overline{X}$ satisfying: (i) $h^-(\delta hc) = \delta c$, where $c \in \{c^+, c^-\}$, $\delta \in H^*$; (ii) $x \leq y \Rightarrow h^-(x) \leq h^-(y)$ [3]. The computational model is developed based on many-valued modus ponens. To guarantee the soundness of modus ponens, the truth function of the implication (called *implicator*) must be *residual* to the t-norm evaluating modus ponens (see [1]). Precisely, let r be the truth value of an implication $h \leftarrow b$, \mathcal{C} a t-norm, and \leftarrow^\bullet its residual implicator; we have:

$$(\forall b)(\forall h) \mathcal{C}(b, \leftarrow^\bullet(h, b)) \leq h \tag{1}$$

$$(\forall b)(\forall r) \leftarrow^\bullet(\mathcal{C}(b, r), b) \geq r \tag{2}$$

Since all values in \overline{X} are linearly ordered, assume they are $v_0 \leq v_1 \leq \dots \leq v_n$, where $v_0 = 0, v_n = 1$. The Łukasiewicz t-norm and implicator can be defined as:

$$\mathcal{C}_L(v_i, v_j) = \begin{cases} v_{i+j-n} & \text{if } i + j - n > 0 \\ v_0 & \text{otherwise} \end{cases}$$

$$\leftarrow_L^\bullet(v_j, v_i) = \begin{cases} v_n & \text{if } i \leq j \\ v_{n+j-i} & \text{otherwise} \end{cases}$$

and those of Gödel can be:

$$\mathcal{C}_G(v_i, v_j) = \min(v_i, v_j)$$

$$\leftarrow_G^\bullet(v_j, v_i) = \begin{cases} v_n & \text{if } i \leq j \\ v_j & \text{otherwise} \end{cases}$$

We can see that t-norms are monotonic in all arguments, and implicators are non-decreasing in the first argument and non-increasing in the second.

2.2 Fuzzy Linguistic Logic Programming

The language is a many sorted (typed) predicate language without function symbols. Connectives can be: conjunctions \wedge, \wedge_L (Łukasiewicz); the disjunction \vee ; implications \leftarrow_L (Łukasiewicz), \leftarrow_G (Gödel); and hedges as unary connectives. For a connective c different from hedges, its truth function is denoted by c^\bullet .

A *term* is either a constant or a variable. An *atom* is of the form $p(t_1, \dots, t_n)$, where p is an n -ary predicate symbol, and t_1, \dots, t_n are terms of corresponding attributes. A *body formula* is defined inductively as follows: (i) An atom is a body formula. (ii) If F and G are body formulae, then so are $\wedge(F, G), \vee(F, G)$ and hF , where h is a hedge (here we use the prefix notation for connectives in body formulae). A *rule* is a graded implication $(A \leftarrow B.r)$, where A is an atom called *rule head*, B is a body formula called *rule body*, and r is a truth value different from 0. $(A \leftarrow B)$ is called the *logical part* of the rule. A *fact* is a graded atom $(A.b)$, where A is an atom called the logical part of the fact, and b is a truth value different from 0. A *query* is an atom used as a question $?A$. A *fuzzy linguistic logic program* (program, for short) is a finite set of rules and facts.

A program P can be represented as a partial mapping:

$$P : \text{Formulae} \rightarrow \overline{X} \setminus \{0\}$$

where the domain of P , $\text{dom}(P)$, is finite and consists only of logical parts of rules and facts, and \overline{X} is a linguistic truth domain. The truth value of a rule $(A \leftarrow B.r)$ is $r = P(A \leftarrow B)$, and that of a fact $(A.b)$ is $b = P(A)$.

As in [4], given a program P , we refer to *Herbrand universe* of a sort A , which consists of all ground terms of A , by U_P^A and *Herbrand base* of P , which consists of all ground atoms, by B_P .

Given a program P , let \overline{X} be the linguistic truth domain; a *fuzzy linguistic Herbrand interpretation* (interpretation, for short) f is a mapping $f : B_P \rightarrow \overline{X}$. We say $f_1 \sqsubseteq f_2$ iff $f_1(A) \leq f_2(A)$ for all $A \in B_P$. The set of all interpretations of a program is a complete lattice under \sqsubseteq . The *bottom interpretation*, denoted \perp , maps every ground atom to 0. An interpretation f can be extended to all formulae, denoted \overline{f} , as follows: (i) $\overline{f}(A) = f(A)$, if A is a ground atom; (ii) $\overline{f}(c(B_1, B_2)) = c^\bullet(\overline{f}(B_1), \overline{f}(B_2))$, where B_1, B_2 are ground formulae, and c is a binary connective; (iii) $\overline{f}(hB) = h^-(\overline{f}(B))$, where B is a ground body formula, and h is a hedge; (iv) $\overline{f}(\varphi) = \text{inf}_{\vartheta} \{\overline{f}(\varphi\vartheta) \mid \varphi\vartheta \text{ is a ground instance of } \varphi\}$. An interpretation f is a *model* of a program P if for all $\varphi \in \text{dom}(P)$, $\overline{f}(\varphi) \geq P(\varphi)$.

Definition 1 (Correct answer [3]). Given a program P , let \overline{X} be the linguistic truth domain. A pair $(x; \theta)$, where $x \in \overline{X}$, and θ is a substitution, is called a correct answer for P and $?A$ if for any model f of P , $\overline{f}(A\theta) \geq x$.

Definition 2 (Admissible rules [3]). Admissible rules are defined as follows:

Rule 1. From $((XA_mY); \vartheta)$ infer $((XC(B, r)Y)\theta; \vartheta\theta)$ if: (i) A_m is an atom (called the selected atom); (ii) θ is an mgu of A_m and A ; (iii) $(A \leftarrow B.r)$ is a rule in the program.

Rule 2. From (XA_mY) infer $(X0Y)$ (this rule is used for situations where A_m does not unify with any rule head or logical part of facts in the program).

Rule 3. (for hedges) From $(XhBY)$ infer $(Xh^-(B)Y)$ if B is a non-empty body formula.

Rule 4. From $((XA_mY); \vartheta)$ infer $((XrY)\theta; \vartheta\theta)$ if: (i) A_m is an atom (also called the selected atom); (ii) θ is an mgu of A_m and A ; (iii) $(A.r)$ is a fact in the program.

Rule 5. If there are no more predicate symbols in the word, replace all connectives \wedge 's and \vee 's with \wedge^\bullet and \vee^\bullet , respectively. Then, since this word contains only some additional C 's, h^- 's and truth values, evaluate it. The substitution remains unchanged.

Definition 3 (Computed answer [3]). Let P be a program and $?A$ a query. A pair $(r; \theta)$, where r is a truth value, and θ is a substitution, is said to be a computed answer if there is a sequence G_0, \dots, G_n such that: (i) $G_0 = (A; id)$; (ii) Every G_{i+1} is inferred from G_i by one of admissible rules (here we also utilise the usual Prolog renaming of variables along derivation); (iii) $G_n = (r; \theta')$ and $\theta = \theta'$ restricted to variables of A . The computation is said to have length n .

Example 1 ([3]). Assume that the truth domain is generated from the 2-limited monotonic HA described in Example 3 in [3], and there is a knowledge base consisting of the following: (i) Sentence “If a student studies very hard, and his/her university is more-or-less high-ranking, then he/she will be a good employee” has a degree of truth “Very Probably True”; (ii) Sentence “The university where Ann is studying is probably high-ranking” has a degree of truth “True”; (iii) Sentence “Ann is studying hard” has a degree of truth “More Very True”.

Let $GdEm$, $StHd$, $HiraUn$, and T stand for “good employee”, “study hard”, “high-ranking university”, and “True”, respectively. Then, the knowledge base can be represented by the following program: (i) $(GdEm(x) \leftarrow \wedge(VStHd(x), MolHiraUn(x)).VPT)$; (ii) $(HiraUn(Ann).PT)$ (note that by Rule $RT1$ [3], from $(PHiraUn(Ann).T)$ we get $(HiraUn(Ann).PT)$); (iii) $(StHd(Ann).MVT)$.

Given a query $?GdEm(Ann)$, we can have the following computation (since the query is ground, the substitution in the computed answer is the identity):

$$\begin{aligned} &?GdEm(Ann) \\ &C(\wedge(VStHd(Ann), MolHiraUn(Ann)), VPT) \\ &C(\wedge(V^-(StHd(Ann)), MolHiraUn(Ann)), VPT) \\ &C(\wedge(V^-(StHd(Ann)), Mol^-(HiraUn(Ann))), VPT) \\ &C(\wedge(V^-(MVT), Mol^-(HiraUn(Ann))), VPT) \\ &C(\wedge(V^-(MVT), Mol^-(PT)), VPT) \\ &C(\wedge^\bullet(V^-(MVT), Mol^-(PT)), VPT) \end{aligned}$$

Here if we use the inverse mappings of hedges built in Example 3 in [3], Lukasiewicz t-norm, and take min as the truth function of the conjunction, we have: $C(\wedge^\bullet(V^-(MVT), Mol^-(PT)), VPT) = PMolT$. Hence, the sentence “Ann will be a good employee” is at least “Probably More-or-less True”.

Theorem 1 (Soundness of the computational model [3]). *Every computed answer for a program P and a query $?A$ is a correct answer.*

3 Fixpoint Semantics

Like in [2], an immediate consequence operator can be defined as follows:

Definition 4 (Immediate consequence operator). *Let P be a program. The operator T_P mapping from interpretations to interpretations is defined as follows. For every interpretation f and every ground atom $A \in B_P$,*

$T_P(f)(A) = \max\{\sup\{C_i(\overline{f}(B), r) : (A \leftarrow_i B.r) \text{ is a ground instance of a rule in } P\}, \sup\{b : (A.b) \text{ is a ground instance of a fact in } P\}\}.$

Since P is function-free, each Herbrand universe U_P^A is finite, so B_P is finite. Hence, for every $A \in B_P$, the number of ground instances of rules whose rule heads match A and the number of ground instances of facts whose logical parts match A are finite. Thus, the suprema in the definition are, in fact, maxima.

Similar to [5], we have the following results.

Theorem 2. *The operator T_P is monotonic.*

Proof. (sketch) Given two interpretations $f_1 \sqsubseteq f_2$, by induction on the structure of body formulae, we have $\overline{f_1}(B) \leq \overline{f_2}(B)$ for all ground body formulae B . Thus, for each ground instance $(A \leftarrow_i B.r)$ of a rule in P , we have $C_i(\overline{f_1}(B), r) \leq C_i(\overline{f_2}(B), r)$. Therefore, $\sup\{C_i(\overline{f_1}(B), r)\} \leq \sup\{C_i(\overline{f_2}(B), r)\}$. Consequently, $T_P(f_1)(A) \leq T_P(f_2)(A)$ for all $A \in B_P$.

Theorem 3. *The operator T_P is continuous.*

Proof. Recall that a mapping $f : L \rightarrow L$, where L is a complete lattice, is said to be continuous if for every directed subset X of L , $f(\sup(X)) = \sup\{f(x) | x \in X\}$.

Let us prove that for each directed set X of interpretations, $T_P(\sup(X)) = \sup\{T_P(f) | f \in X\}$. Since T_P is monotonic, $\sup\{T_P(f) | f \in X\} \sqsubseteq T_P(\sup(X))$. On the other hand, since B_P and the truth domain are finite, the set of all Herbrand interpretations of P is also finite. Therefore, for each finite directed set X of interpretations, we have an upper bound of X in X . This together with the monotonicity of T_P leads to $T_P(\sup(X)) \sqsubseteq \sup\{T_P(f) : f \in X\}$.

Theorem 4. *An interpretation f is a model of a program P iff $T_P(f) \sqsubseteq f$.*

Proof. First, assume that f is a model of P ; we show that $T_P(f) \sqsubseteq f$. Let $A \in B_P$. Consider the following: (i) If A is neither a ground instance of a logical part of facts nor that of a rule head in P , then $T_P(f)(A) = 0 \leq f(A)$; (ii) For each ground instance $(A.b)$ of a fact, say $(C.b)$, in P , since f is a model of P , and A is a ground instance of C , we have $b = P(C) \leq \overline{f}(C) \leq f(A)$. Hence, $f(A) \geq \sup\{b | (A.b) \text{ is a ground instance of a fact in } P\}$; (iii) For each ground instance

$(A \leftarrow_i B.r)$ of a rule, say $(C.r)$, in P , we have: $\mathcal{C}_i(\bar{f}(B), r) = \mathcal{C}_i(\bar{f}(B), P(C)) \leq \mathcal{C}_i(\bar{f}(B), \bar{f}(A \leftarrow_i B)) = \mathcal{C}_i(\bar{f}(B), \leftarrow_i^\bullet (f(A), \bar{f}(B))) \leq^{(*)} f(A)$, where $(*)$ follows from [\(III\)](#). Therefore, $f(A) \geq \sup\{\mathcal{C}_i(\bar{f}(B), r) \mid (A \leftarrow_i B.r) \text{ is a ground instance of a rule in } P\}$. Thus, by definition $T_P(f)(A) \leq f(A)$ for all $A \in B_P$.

Now we show that if $T_P(f) \sqsubseteq f$, f is a model of P . Let C be any formula in $\text{dom}(P)$. Consider two cases: (i) $(C.c)$, where c is a truth value, is a fact in P . For each ground instance A of C , by hypothesis and definition, $f(A) \geq T_P(f)(A) \geq \sup\{b \mid (A.b) \text{ is a ground instance of a fact in } P\} \geq c = P(C)$. Thus, $\bar{f}(C) = \inf\{f(A) \mid A \text{ is a ground instance of } C\} \geq P(C)$; (ii) $(C.c)$ is a rule in P . For each ground instance $A \leftarrow_j D$ of C , by hypothesis and definition, $f(A) \geq T_P(f)(A) \geq \sup\{\mathcal{C}_i(\bar{f}(B), r) \mid (A \leftarrow_i B.r) \text{ is a ground instance of a rule in } P\} \geq \mathcal{C}_j(\bar{f}(D), c) = \mathcal{C}_j(\bar{f}(D), P(C))$. Hence, $\bar{f}(A \leftarrow_j D) = \leftarrow_j^\bullet (f(A), \bar{f}(D)) \geq \leftarrow_j^\bullet (\mathcal{C}_j(\bar{f}(D), P(C)), \bar{f}(D)) \geq^{(*)} P(C)$, where $(*)$ follows from [\(II\)](#). Consequently, $\bar{f}(C) = \inf\{\bar{f}(A \leftarrow_j D) \mid (A \leftarrow_j D) \text{ is a ground instance of } C\} \geq P(C)$.

Since the given immediate consequence operator T_P satisfies [Theorem 3](#) and [Theorem 4](#), due to Knaster and Tarski [\[8\]](#), the Least Herbrand model of a program P is exactly the least fixpoint of T_P and can be obtained by iterating T_P from the bottom interpretation after ω iterations, where ω is the smallest limit ordinal (apart from 0). Moreover, since the truth domain \bar{X} and Herbrand base B_P are finite, the least model of the program P can be obtained after at most $\mathcal{O}(|P| |\bar{X}|)$ steps, where $|S|$ denotes the cardinality of set S . This is an important tool for dealing with recursive programs, for which computations can be infinite.

4 Completeness of the Computational Model

Theorem 5. *Let P be a program and A a ground atom. For all n , there exists a computation for P and $?A$ such that the computed answer is $(T_P^n(\perp)(A); id)$.*

Proof. Since A is ground, substitutions are the identities. The proof is by induction on n . Suppose first that $n = 0$. $T_P^0(\perp)(A) = 0$, and there is a computation for P and $?A$ in which only Rule 2 is applied with a computed answer $(0; id)$.

Now suppose that the result holds for $n - 1$, where $n \geq 1$. We prove that it also holds for n . There are two cases: (i) A does not unify with any rule head or logical part of facts. Then, $T_P^n(\perp)(A) = 0$, and the computation is the same as that of the case $n = 0$; (ii) Otherwise, by definition of T_P and the fact that the suprema are maxima, there exists either a ground instance $(A.b)$ of a fact such that $T_P^n(\perp)(A) = b$ or a ground instance $(A \leftarrow_i B.r)$ of a rule such that $T_P^n(\perp)(A) = \mathcal{C}_i(T_P^{n-1}(\perp)(B), r)$. For the former case, there is a computation for P and A in which only Rule 1 is applied with a computed answer $(b; id)$. For the latter, by the induction hypothesis, for each ground atom B_j in B there exists a computation such that $T_P^{n-1}(\perp)(B_j)$ is the computed truth value of B_j . Thus, the computed truth value of the whole body B is $T_P^{n-1}(\perp)(B)$ calculated from all $T_P^{n-1}(\perp)(B_j)$ along the complexity of B . Clearly, there exists a computation for P and $?A$ in which the first rule to be applied is Rule 1 carried out on the rule

in P which has $(A \leftarrow_i B.r)$ as its ground instance, and the rest is a combination of computations of all B_j in B with a computed truth value $T_P^n(\perp)(A)$.

The following theorem shows the completeness for the case of ground queries.

Theorem 6. *For every correct answer $(x; id)$ of a program P and a ground query $?A$, there exists a computed answer $(r; id)$ such that $r \geq x$.*

Proof. Since $(x; id)$ is a correct answer of P and $?A$, for every model f of P we have $f(A) \geq x$. In particular, let M_P be the Least Herbrand model of P ; $M_P(A) = T_P^w(\perp)(A) \geq x$. Recall that $T_P^w(\perp)(A) = \sup\{T_P^n(\perp)(A) | n < w\}$. Since w is a finite number, the *sup* operator is, in fact, a maximum. Hence, there exists $n < w$ such that $T_P^n(\perp)(A) = T_P^w(\perp)(A)$. By Theorem 5, there exists a computation for P and $?A$ such that the computed answer is $(T_P^n(\perp)(A); id)$.

For the case of non-ground queries, we need to define several more notions. Given a computation with length n for a program P and a query $?A$, we call each $G_i, i = 0 \dots (n-1)$, in the sequence of the computation, an *intermediate query* (IQ) and the part of the computation from G_i to G_n an *intermediate computation* (IC) with length $n - i$. Thus, a computation is a special case of ICs with $i = 0$. Furthermore, we define an *unrestricted computation* (UC) (*unrestricted intermediate computation* (UIC)) as a computation (intermediate computation) in which the substitutions θ_i in each step are not necessary to be most general unifiers, but only required to be unifiers.

Mgu and Lifting lemmas in [4] can be extended for FLLP as follows.

Lemma 1 (Extended mgu lemma). *Let P be a program and G_i for some i an IQ. Suppose that there exists an UIC for P and G_i . Then there exists an IC for P and G_i with the same computed truth value and length such that, if $\theta_{i+1}, \dots, \theta_n$ are the unifiers from the UIC and $\theta'_{i+1}, \dots, \theta'_n$ are the mgu's from the IC, then there exists a substitution γ such that $\theta_{i+1} \dots \theta_n = \theta'_{i+1} \dots \theta'_n \gamma$.*

Proof. The proof is by induction on the length k of the UIC.

Suppose first that $k = 1$, i.e., $n = i + 1$. It is easily verified that only Rule 4 can be the last rule to be applied in an UIC, and furthermore the UIC is, in fact, an UC with length 1. That means $i = 0, G_0 = (A; id)$, where A is an atom, and there exists a fact $(A_m.b)$ in P such that θ_1 is a unifier of A and A_m , and b is the computed truth value. Assume that θ'_1 is an mgu of A and A_m . Then, $\theta_1 = \theta'_1 \gamma$ for some γ . Clearly, there is a computation for P and $?A$ carried out on the same fact with length 1, computed truth value b , and mgu θ'_1 .

Suppose that the result holds for length $\leq k - 1$, where $k \geq 2$; we will prove that it also holds for length k . Consider the transition from G_i to G_{i+1} . Since $k \geq 2$, it can not be an application of Rule 5 and thus is one of the following:

(i) Either Rule 2 or Rule 3 is applied. Then, $\theta_{i+1} = id$. By the induction hypothesis, there exists an IC for P and G_{i+1} of length $k - 1$ with mgu's $\theta'_{i+2}, \dots, \theta'_n$ such that $\theta_{i+2} \dots \theta_n = \theta'_{i+2} \dots \theta'_n \gamma$ for some γ . Thus, there is an IC for P and G_i with mgu's $\theta'_{i+1} = id, \theta'_{i+2}, \dots, \theta'_n$ and $\theta_{i+1} \dots \theta_n = \theta'_{i+1} \dots \theta'_n \gamma$.

(ii) Either Rule 1 or Rule 4 is applied. Hence, θ_{i+1} is a unifier of the selected atom A in G_i and an atom A' , which is either a rule head or a logical part of a fact in P . There exists an mgu θ'_{i+1} of A and A' such that $\theta_{i+1} = \theta'_{i+1}\vartheta$ for some ϑ . Now, if we use θ'_{i+1} instead of θ_{i+1} in the transition, we will obtain an IQ G'_{i+1} and $G_{i+1} = G'_{i+1}\vartheta$. Consider all transitions from G_{i+1} to G_{n-1} . Since they can not be an application of Rule 5, there are two cases:

(a) All the transitions use either Rule 2 or Rule 3. Thus, all unifiers are the identity. If we apply the same rule on the corresponding atom (for the case of Rule 2) or on the corresponding body formula (for the case of Rule 3) for each transition from the IQ G'_{i+1} , we obtain a sequence $G'_{i+1}, \dots, G'_{n-1}$ such that for all $i+1 \leq l \leq n-1$, $G_l = G'_l\vartheta$. Since the transition from G_{n-1} to G_n uses Rule 5, G_{n-1} does not contain any predicate symbols, and neither does G'_{n-1} , so they are identical. As a consequence, P and G_i have an IC $G_i, G'_{i+1}, \dots, G'_{n-1}, G_n$ with mgu's being θ'_{i+1} and the identities.

(b) There exists the smallest m such that $i+1 \leq m \leq n-2$, and the transition from G_m to G_{m+1} uses either Rule 1 or Rule 4. We will prove the result for the case of Rule 1, and the case of Rule 4 can be proved similarly. Since all transitions from G_{i+1} to G_m use either Rule 2 or Rule 3, we can have a sequence G'_{i+1}, \dots, G'_m such that for all $i+1 \leq l \leq m$, $G_l = G'_l\vartheta$. The application of Rule 1 for the transition from G_m to G_{m+1} implies that there exists a rule $(A'' \leftarrow_j B.r)$ in P such that θ_{m+1} is a unifier of the selected atom A_m in G_m and A'' . Suppose that A'_m is the corresponding selected atom in G'_m , we have $A_m = A'_m\vartheta$. Since we utilise the usual Prolog renaming of variables along derivation, we can assume that ϑ does not act on any variables of A'' or B , so $\vartheta\theta_{m+1}$ is a unifier of A'_m and A'' . Now applying Rule 1 for G'_m on the selected atom A'_m and rule $(A'' \leftarrow_j B.r)$ with unifier $\vartheta\theta_{m+1}$, we obtain G'_{m+1} , and it can be shown that $G'_{m+1} = G_{m+1}$. Thus P and G'_m have an UIC $G'_m, G_{m+1}, \dots, G_n$ with unifiers $\vartheta\theta_{m+1}, \theta_{m+2}, \dots, \theta_n$. By the induction hypothesis, P and G'_m have an IC having the same computed truth value with mgu's $\theta'_{m+1}, \dots, \theta'_n$ such that $\vartheta\theta_{m+1}\theta_{m+2}\dots\theta_n = \theta'_{m+1}\dots\theta'_n\gamma$ for some γ . Since $\theta_{i+2}, \dots, \theta_m$ are the identity, P and G_i have an IC $G_i, G'_{i+1}, \dots, G'_m, G'_{m+1}, \dots, G'_n$ with mgu's $\theta'_{i+1}, \theta_{i+2}, \dots, \theta_m, \theta'_{m+1}, \dots, \theta'_n$ and $\theta_{i+1}\dots\theta_n = \theta'_{i+1}\theta_{i+2}\dots\theta_m\vartheta\theta_{m+1}\theta_{m+2}\dots\theta_n = \theta'_{i+1}\theta_{i+2}\dots\theta_m\theta'_{m+1}\dots\theta'_n\gamma$.

Lemma 2 (Extended lifting lemma). *Let P be a program, $?A$ a query, and θ a substitution. Suppose there is a computation for P and $?A\theta$. Then there exists a computation for P and $?A$ with the same length and computed truth value such that, if $\theta_1, \dots, \theta_n$ are mgu's from the computation for P and $?A\theta$, and $\theta'_1, \dots, \theta'_n$ are mgu's from the computation for P and $?A$, then there exists a substitution γ such that $\theta\theta_1\dots\theta_n = \theta'_1\dots\theta'_n\gamma$.*

Proof. Suppose the sequence of the computation for P and $?A\theta$ is $G_0 = (A\theta; id), G_1, \dots, G_n$. Consider the admissible rule applied in the transition from G_0 to G_1 . We will prove for the case of Rule 1, and it can be proved similarly for the others. The application of Rule 1 implies that there exists a rule $(A' \leftarrow_j B.r)$ in P such that θ_1 is an mgu of $A\theta$ and A' . We can assume that θ does not act on any variables of A' or B . Thus, $\theta\theta_1$ is a unifier of A and A' . Applying Rule 1 for

$G'_0 = (A; id)$ on rule $(A' \leftarrow_j B.r)$ with unifier $\theta\theta_1$, we obtain $G'_1 = G_1$. Hence, we have an UC for P and $?A$, which looks like the given computation for P and $?A\theta$, except that the first IQ G'_0 is different, and the first unifier is $\theta\theta_1$. Now applying the extended mgu lemma, we obtain the result.

Lemma 3. *Let P be a program and $?A$ a query. Suppose $(x; \theta)$ is a correct answer for P and $?A$. Then there exists a computation for P and $?A\theta$ with a computed answer $(r; id)$ such that $r \geq x$.*

Proof. The proof is similar to that of Lemma 8.5 in [4]. Suppose that $A\theta$ has variables x_1, \dots, x_n . Let a_1, \dots, a_n be distinct constants not appearing in P or A , and let θ_1 be the substitution $\{x_1/a_1, \dots, x_n/a_n\}$. Since for any model f of P , $\overline{f}(A\theta\theta_1) \geq \overline{f}(A\theta) \geq x$, and $A\theta\theta_1$ is ground, $(x; id)$ is a correct answer for P and $?A\theta\theta_1$. By Theorem 6, there exists a computation for P and $?A\theta\theta_1$ with a computed answer $(r; id)$ such that $r \geq x$. Since the a_i do not appear in P or A , by replacing a_i with x_i ($i = 1, \dots, n$) in this computation, we obtain a computation for P and $?A\theta$ with the computed answer $(r; id)$.

Theorem 7 (Completeness of the computational model). *Let P be a program and $?A$ a query. For every correct answer $(x; \theta)$ for P and $?A$, there exists a computed answer $(r; \sigma)$ and a substitution γ such that $r \geq x$ and $\theta = \sigma\gamma$.*

Proof. Since $(x; \theta)$ is a correct answer for P and $?A$, by Lemma 3, there exists a computation for P and $?A\theta$ with a computed answer $(r; id)$ such that $r \geq x$. Suppose the sequence of mgu's of the computation is $\theta_1, \dots, \theta_n$. Then $A\theta\theta_1 \dots \theta_n = A\theta$. By the extended lifting lemma, there exists a computation for P and $?A$ with the same computed truth value r and mgu's $\theta'_1, \dots, \theta'_n$ such that $\theta\theta_1 \dots \theta_n = \theta'_1 \dots \theta'_n \gamma'$ for some γ' . Let σ be $\theta'_1 \dots \theta'_n$ restricted to variables in A . Then $\theta = \sigma\gamma$, where γ is an appropriate restriction of γ' .

5 Conclusion

In this paper, we have presented a fixpoint semantics for fuzzy linguistic logic programs and based on it proved the completeness of the computational model.

References

1. Hájek, P.: Metamathematics of Fuzzy Logic. Kluwer Academic Publishers, Dordrecht (1998)
2. Krajčí, S., Lencses, R., Vojtáš, P.: A Comparison of Fuzzy and Annotated Logic Programming. Fuzzy Sets and Syst. 144, 173–192 (2004)
3. Le, V.H., Liu, F., Tran, D.K.: Fuzzy Linguistic Logic Programming. In: Huang, D.S., et al. (eds.) ICIC 2008. LNCS (LNAI) (to appear, 2008)
4. Lloyd, J.W.: Foundations of Logic Programming. Springer, Berlin (1987)
5. Medina, J., Ojeda-Aciego, M., Vojtáš, P.: Similarity-Based Unification: A Multi-adjoint Approach. Fuzzy Sets and Syst. 146, 43–62 (2004)

6. Nguyen, C.H., Wechler, W.: Hedge Algebras: An Algebraic Approach to Structure of Sets of Linguistic Truth Values. *Fuzzy Sets and Syst.* 35, 281–293 (1990)
7. Nguyen, C.H., Wechler, W.: Extended Hedge Algebras and their Application to Fuzzy Logic. *Fuzzy Sets and Syst.* 52, 259–281 (1992)
8. Tarski, A.: A Lattice-Theoretical Fixpoint Theorem and its Applications. *Pacific J. of Math.* 5, 285–309 (1955)
9. Vojtáš, P.: Fuzzy Logic Programming. *Fuzzy Sets and Syst.* 124, 361–370 (2001)

Forest-RK: A New Random Forest Induction Method

Simon Bernard, Laurent Heutte, and Sébastien Adam

Université de Rouen, LITIS EA 4108

BP 12 - 76801 Saint-Etienne du Rouvray, France.

{simon.bernard, laurent.heutte, sebastien.adam}@univ-rouen.fr

Abstract. In this paper we present our work on the parametrization of Random Forests (RF), and more particularly on the number K of features randomly selected at each node during the tree induction process. It has been shown that this hyperparameter can play a significant role on performance. However, the choice of the value of K is usually made either by a greedy search that tests every possible value to choose the optimal one, either by choosing *a priori* one of the three arbitrary values commonly used in the literature. With this work we show that none of those three values is always better than the others. We thus propose an alternative to those arbitrary choices of K with a new “push-button” RF induction method, called Forest-RK, for which K is not an hyperparameter anymore. Our experimentations show that this new method is at least as statistically accurate as the original RF method with a default K setting.

Keywords: Classification; Classifier Ensemble; Classifier Combination; Random Forests; Decision Trees; Bagging.

1 Introduction

Random Forest is a family of classifier ensemble methods that use randomization to produce a diverse pool of individual classifiers, as for Bagging [1] or Random Subspaces methods [2]. It can be defined as a generic principle of classifier combination that uses L tree-structured base classifiers $\{h(x, \Theta_k), k = 1, \dots, L\}$ where $\{\Theta_k\}$ is a family of independent identically distributed random vectors, and x is an input data. The particularity of this kind of ensemble is that each decision tree is built from a random vector of parameters. A Random Forest can be built for example by randomly sampling a feature subset for each decision tree (as in Random Subspaces), and/or by randomly sampling a training data subset for each decision tree (as in Bagging). Since they have been introduced in 2001, RFs have been studied in many ways, both theoretically and experimentally [3,4,5,6,7,8,9,10,11]. In most of those works, it has been shown that RFs are particularly competitive with one of the most efficient learning principles, i.e. boosting [5,7,10]. However, the mechanisms that explain the good performance of this type of classifier ensemble are not clearly identified and one has to admit

that it is still a complex task for the practitioner to take full benefits of the potential of those methods. For example considering the reference RF method called Forest-RI, introduced by Breiman in [5] (cf. section 2), an important hyperparameter has been identified : the number K of features randomly selected at each node during the tree induction process. Yet, in those research works that have experimented this method, the value of K is arbitrarily or empirically set, and sometimes without any theoretical nor experimental justification.

In this paper we propose an alternative to those arbitrary settings of K . Indeed, it appears that this hyperparameter setting is crucial for accuracy in RF induced with Forest-RI algorithm [3]. However, the choice of the value of K is usually made either by a greedy search that tests every possible value to choose the optimal one, either by choosing *a priori* one of the three arbitrary values commonly used in the literature. With this work we show that none of those three values is always better than the others and that none of them consequently represent a good setting. We thus propose a new RF algorithm, called Forest-RK, based on Forest-RI but for which the setting of K does not play a crucial role anymore for growing accurate RF classifiers. We show that this new method is at least as statistically accurate as the “classic” Forest-RI algorithm with default settings.

The paper is thus organized as follows: in the following section, we detail the Forest-RI algorithm used in our experiments; in section 3, we describe our experimental protocol, the datasets used, and compare the results obtained with different settings of the hyperparameter K . We finally draw some conclusions and future works in the last section.

2 The Forest-RI Algorithm

One can see RFs as a family of methods, made of different decision tree ensemble induction algorithms, such as the Breiman Forest-RI method often cited as the reference algorithm in the literature [5]. In this algorithm the Bagging principle is used with another randomization technique called Random Feature Selection. The training step consists in building an ensemble of decision trees, each one trained from a bootstrap sample of the original training set — i.e. applying the Bagging principle — and with a decision tree induction method called Random Tree. In this induction algorithm, a feature subset is randomly drawn for each node, from which the best splitting criterion is then selected. Thus, the Forest-RI method grows a decision tree using the following process :

- Let N be the size of the original training set. N instances are randomly drawn with replacement, to form the bootstrap sample, which is then used to build a tree.
- Let M be the dimensionality of the original feature space, and K a preliminary fixed parameter so that $K \in [1, M]$. For each node of the tree, a subset of K features is randomly drawn without replacement, among which the best split is then selected.
- The tree is thus built to reach its maximum size. No pruning is performed.

This process is thus led by an important hyperparameter: the number K of randomly selected features in the splitting process. Whereas this parameter has already shown to be critical for RF performance, no research work has been specifically devoted to study its setting and its real influence on performance, and only a few have empirically dealt with this issue.

In [8] for example, Guerts et al. have proposed a new method of RF induction, called Extras-Trees for Extremely Randomized Tree Ensemble, that modifies the Forest-RI algorithm to accentuate the randomization. Here the Random Feature Selection is still used but modified so that the best splitting criterion selection is one step further randomized. The authors have designed their experimental protocol to study the influence of K on performance. Even if this method is partly different from the Forest-RI algorithm, this work allows to draw some intuitions on the RF behavior according to K . It highlights for example that their default setting $K = \sqrt{M}$, where M stands for the dimensionality of the original feature space, is most of times closed to the optimal setting, at least for the Extras-Trees method and on several representative datasets. Another example is Breiman’s work on performance according to this K parameter [5]. In these experiments, a large number of RF has been grown on three databases of the UCI repository, for which the test set error rate has been monitored. Actually only one of those three experiments was really concerned by the Forest-RI algorithm, since the two others have been run with a different induction algorithm that uses feature combinations in the splitting criterion, instead of single features. Hence, even if Breiman draws some tendencies [5], those experiments do not allow to conclude on RF behavior according to the setting of K . We also noticed that in his Forest-RI experiments, Breiman has decided to use two values of K : 1 and $\log_2(M) + 1$. While the first value is intuitively interesting since it corresponds to a decision tree induction that selects in a fully random manner the splitting criterion among features for each node, the second one seems to be more arbitrary or at least is not justified.

Finally, implementation and experimentation of the Forest-RI algorithm require to fix the value of the hyperparameter K but as we have shown, there actually does not exist any theoretical rule that can be used to fix it. As mentioned previously, only arbitrary default values are proposed in the literature and nothing guarantees that these values are close to the optimal setting, as we will show in section section 3.3. We thus propose an alternative to those settings, with a new push-button RF induction algorithm for which K is not an hyperparameter anymore. We describe in the following section our new algorithm and compare it with the “classic” Forest-RI algorithm, through extensive experiments on several datasets and with predefined values of K .

3 Investigating the Influence of K on Forest-RI Performance

The purpose of this set of experiments is to compare accuracies of Forest-RI algorithm with default settings of K on the one hand, to our new push-button

RF algorithm in which K is not an hyperparameter anymore, on the other hand. In this new algorithm, called Forest-RK, K still exists since Random Feature Selection is still used, but is randomly chosen for each splitting node. Instead of fixing the value of K so that it is identical for all the decision trees, a new value of K is randomly chosen at each node of the trees, and used for this current node splitting only. The new Forest-RK decision tree induction procedure can be summarized as below:

- Let N be the size of the original training set. N instances are randomly drawn with replacement, to form the bootstrap sample, which is then used to build a tree.
- Let M be the dimensionality of the original feature space. Randomly set a number $K \in [1, M]$ for each node of the tree, so that a subset of K features is randomly drawn without replacement, among which the best split is then selected.
- The tree is thus built to reach its maximum size. No pruning is performed.

As shown in the above algorithm, the main difference between Forest-RI and Forest-RK lies in that K is randomly chosen for each node of the tree leading therefore to more diverse trees in the Forest-RK than in the Forest-RI. Two ideas have led us to this new algorithm: i) it avoids the greedy iterative test of every possible value of K to find the best one, while being at least as accurate as the traditional Forest-RI algorithm parametrized with default values; ii) it is an alternative to default settings of K , since as shown in section 3.3, those values are arbitrary and are not always the best choice. Thus the problem is now to determine which of the two algorithms, Forest-RI or Forest-RK, will produce more accurate classifiers when trained and tested on the same datasets. To answer to this question, *i.e.* assessing which of the two algorithms performs better, we lean on the comparison of five approximate statistical tests, compared in 12. In this paper, it is recommended to use McNemar’s test 13, for which it is shown that it better suits to experimental protocols like ours. The McNemar’s test is firstly used here to determine whether or not two sets of predictions differ significantly. That is to say, under the null hypothesis H_0 , the two algorithms should have the same error rate. Given two algorithms A and B producing two classifiers h_A and h_B , the contingency table is constructed, so that it gives the four values : $n_{00} = \#$ samples misclassified by both h_A and h_B ; $n_{01} = \#$ samples misclassified by h_A but not by h_B ; $n_{10} = \#$ samples misclassified by h_B but not by h_A and $n_{11} = \#$ samples misclassified by neither h_A nor h_B . McNemar’s test is then based on a χ^2 test for goodness-of-fit that compares the distribution of counts expected under the null hypothesis to the observed counts. It thus states that the statistics X^2 (equation 1) can be considered as following a χ^2 distribution with 1 degree of freedom.

$$X^2 = \frac{(|n_{01} - n_{10}| - 1)^2}{n_{01} + n_{10}} \sim \chi_{1,0.05}^2 = 3.841459 \quad (1)$$

Consequently H_0 is rejected, *i.e.* one of the two algorithms is considered to be “better” than the other, if X^2 is greater than $\chi_{1,0.05}^2 = 3.841459$. Finally, when

applied on a specific testing set, three answers can be obtained through the McNemar test :

- H_0 is rejected and $n_{01} > n_{10}$: Algorithm B produces significantly more accurate classifiers than algorithm A .
- H_0 is rejected and $n_{01} < n_{10}$: Algorithm A produces significantly more accurate classifiers than algorithm B .
- H_0 is accepted: The two algorithms do not produce classifiers significantly different in term of accuracy.

With such a procedure, we are able to assess if Forest-RK statistically outperforms or not the Forest-RI algorithm with default K parameter settings. We first describe in the following subsection the datasets used. We then detail our experimental protocol to compare the two algorithms and discuss the obtained results.

3.1 Datasets

The 10 datasets that have been used for these experimentations are described in table 1. The first 8 datasets have been selected from the UCI repository [14]; the two last, Twonorm and Ringnorm, are synthetic datasets designed by Breiman [15]. Those datasets have firstly been selected because they are representative of typical machine learning issues in terms of number of classes, of features and of samples. They have also been chosen because they do not contain any missing values and because the features are all numerical. All those datasets are not preliminary divided into training and testing subsets. Thus for our experiments we have decided to randomly split each original dataset, with two thirds of the instances used for training, and the other third for testing. In order to make sure that our results do not depend on this arbitrary splitting this process has been repeated 10 times with various splittings.

Table 1. Datasets description

Dataset	Size	Features	Classes	Dataset	Size	Features	Classes
Diabetes	768	8	2	Spambase	4610	57	2
Gamma	19020	10	2	Vehicle	946	18	4
Letter	20000	16	26	Waveform	5000	40	3
Pendigits	10992	16	10	Ringnorm	7400	20	2
Segment	2310	19	7	Twonorm	7400	20	2

3.2 Experimental Protocol

In this section our experimental protocol is described. It performs comparative tests on the datasets detailed in table 1. For all the experiments described in this section, the number of trees grown in the forests has been set to 100. This choice is based on a previous experimental work presented in [3], in which we have shown that it is a reasonable value to grow an accurate RF, and considering

that we do not seek to reach intrinsic optimal performance. First, each dataset has been randomly split into training and testing subsets, as explained in the previous section. This splitting procedure has been repeated 10 times so that 10 different training sets and testing sets are thus available, each set containing two thirds and one third of the original dataset respectively. We denote by $T_i = (Tr_i, Ts_i)$ such a split, with $i \in [1, 10]$ and where Tr_i and Ts_i stand respectively for the training part and the test part. Then, for each T_i , the Forest-RI algorithm has been run with the three following default values of K : $K = 1$, $K = \sqrt{M}$ and $K = \log_2(M) + 1$; and Forest-RK has been run 50 times. By this way, $10 \times 50 \times 3 = 1500$ comparisons between Forest-RI and Forest-RK have been performed for each dataset. Algorithm 1 summarizes the whole experimental protocol applied to each dataset. This procedure outputs for each dataset 3 tables of 500 McNemar test outputs, i.e. values $\in \{-1, 0, 1\}$, corresponding to the three possible cases enumerated previously. Those results are presented and discussed in the next section.

Algorithm 1. Experimental Protocol

Input: N the number of instances in the original dataset.

Input: M the number of features in the original dataset.

for $i = 1$ **to** 10 **do**

Randomly draw without replacement $\frac{2}{3} \times N$ of the original dataset instances to form a training subset Tr_i . The remaining instances form the test subset Ts_i , and the couple (Tr_i, Ts_i) is denoted T_i .

Grow three Random Forests, on the training set Tr_i , noted h_1 , $h_{\sqrt{M}}$ and $h_{\log_2(M)+1}$, according to Forest-RI algorithm with K respectively equal to 1, \sqrt{M} and $\log_2(M) + 1$.

for $j = 1$ **to** 50 **do**

Grow a Random Forest, noted h_{R_j} , according to Forest-RK algorithm, on the training set Tr_i .

Apply McNemar test on classifier pairs (h_1, h_{R_j}) , $(h_{\sqrt{M}}, h_{R_j})$ and $(h_{\log_2(M)+1}, h_{R_j})$, with the testing set Ts_i . Store the results.

end for

end for

3.3 Results

Table 2 presents a synthesis of our results obtained by the experimental protocol detailed in Algorithm 1. As mentioned above, 3 tables of 500 comparison results are firstly obtained for each dataset. For those 3 tables, the number of occurrences of the three possible cases have been counted and detailed in table 2. The first observation that can be made from this table is that, when all the results are summed for each of the three McNemar possible answers, the second case for which the two algorithms have shown to be statistically equivalent, is strongly in the majority. Thus, considering all the comparisons performed between Forest-RI and Forest-RK, the two algorithms are as accurate as each other. Then looking

at each cell of the table the same observation can be made : for each case — except for two of them, i.e. Letter/ $K = 1$ and Ringnorm/ $K = 1$ — the two algorithms have shown to be equivalent most of the times. Hence, the McNemar test indicates that the two algorithms are “globally” equivalent; however, as the first and third values in each cell of the table are not always strictly null, one can say that one of the two algorithms is sometimes better than the other. Thus, let us consider only cases for which the McNemar test indicates that one algorithm outperforms the other, by comparing the first and the third values in each cell of the table: we can notice that in 19 of the 30 duels, Forest-RI outperforms Forest-RK most of times, and in 9 of the 30 duels it is the contrary. But concerning the Forest-RI algorithm, when looking at each dataset, it appears to be more interesting to use $K = 1$ for 3 datasets (Spambase, Ringnorm, Twonorm), to use $K = \sqrt{M}$ for 4 datasets (Pendigits, Gamma, Letter, Segment) and to use $K = \log_2(M) + 1$ for 3 datasets (Diabetes, Vehicle, Waveform). Finally, choosing the best value of K in Forest-RI algorithm still remains an unsolved problem since it depends on the intrinsic characteristics of the tested dataset. Consequently the Forest-RK algorithm is a good alternative to Forest-RI for producing accurate Random Forest classifiers since it provides a means to avoid the selection of the optimal value of K while providing the same good performance in average than those obtained with the best value of K .

Table 2. McNemar test results. In each cell the first number corresponds to cases for which Forest-RK outperforms Forest-RI; the second number to cases for which neither of the two algorithms outperforms the other; and the third number to cases for which Forest-RI outperforms Forest-RK. The number in brackets represents the corresponding value of K .

Dataset	$K = 1$	$K = \sqrt{M}$	$K = \log_2(M) + 1$
Diabetes	4/482/14	0/490/10 (3)	12/471/17 (4)
Gamma	3/495/2	8/473/19 (3)	6/488/6 (4)
Letter	425/75/0	1/350/149 (4)	0/374/126 (5)
Pendigits	38/467/0	4/491/5 (4)	49/451/0 (5)
Segment	142/358/0	7/490/3 (4)	6/494/0 (5)
Spambase	1/408/91	0/416/84 (8)	0/452/48 (7)
Vehicle	5/495/0	6/491/3 (4)	5/490/5 (5)
Waveform	50/450/0	1/447/52 (6)	0/440/60 (6)
Ringnorm	0/3/497	0/414/86 (4)	1/479/20 (5)
Twonorm	0/296/204	1/485/14 (4)	1/483/16 (5)
Sums	627/3034/839	20/4074/406	74/4134/292

4 Conclusions

Investigations on RF parametrization have been presented in this paper, that have focused on the number K of features randomly selected at each node during the tree induction process. A new push-button algorithm has been presented

for which the setting of K is not a crucial issue anymore. Experimental comparisons with the reference algorithm Forest-RI, using the McNemar statistical test of significance, have shown that this new algorithm produces classifiers that are statistically as accurate as the Forest-RI induced RF with default settings of K usually found in the literature. Since the setting of K for the Forest-RI algorithm is still an unsolved issue, it appears that Forest-RK is a good alternative for producing accurate classifiers. However, some issues still remain when focusing on hyperparameter K : Is there any other value than those proposed in the literature, that can make Forest-RI produce more accurate classifiers? How can the optimal setting be determined? Answering to those questions would furthermore be interesting for adapting this new algorithm so that it would be able to change the setting of K to a value known to be useful to produce a more accurate classifier.

References

1. Breiman, L.: Bagging Predictors. *Machine Learning* 24(2), 123–140 (1996)
2. Ho, T.: The Random Subspace Method for Constructing Decision Forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20(8), 832–844 (1998)
3. Bernard, S., Heutte, L., Adam, S.: Using Random Forests for Handwritten Digit Recognition. In: *International Conference on Document Analysis and Recognition*, pp. 1043–1047 (2007)
4. Boinee, P., Angelis, A.D., Foresti, G.: Meta Random Forests. *International Journal of Computational Intelligence* 2(3), 138–147 (2005)
5. Breiman, L.: Random Forests. *Machine Learning* 45(1), 5–32 (2001)
6. Breiman, L.: Consistency of Random Forests and Other Averaging Classifiers. Technical Report (2004)
7. Cutler, A., Zhao, G.: Pert - Perfect Random Tree Ensembles. *Computing Science and Statistics* 33 (2001)
8. Geurts, P., Ernst, D., Wehenkel, L.: Extremely Randomized Trees. *Machine Learning* 36(1), 3–42 (2006)
9. Latinne, P., Debeir, O., Decaestecker, C.: Limiting the Number of Trees in Random Forests. In: *2nd International Workshop on Multiple Classifier Systems*, pp. 178–187 (2001)
10. Rodriguez, J., Kuncheva, L., Alonso, C.: Rotation forest: A New Classifier Ensemble Method. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28(10) (2006)
11. Robnik-Sikonja, M.: Improving Random Forests. In: *CSL 2004. LNCS (LNAI)*, vol. 3210, pp. 359–370. Springer, Berlin (2004)
12. Dietterich, T.: Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation* 10, 1895–1923 (1998)
13. Everitt, B.: *The Analysis of Contingency Tables*. Chapman and Hall, London (1977)
14. Asuncion, A., Newman, D.: UCI Machine Learning Repository (2007)
15. Breiman, L.: Arcing classifiers. *The Annals of Statistics* 26(3), 801–849 (1998)

Fuzzy Linguistic Logic Programming

Van Hung Le¹, Fei Liu¹, and Dinh Khang Tran²

¹ Department of Computer Science and Computer Engineering

La Trobe University

Bundoora, VIC 3086, Australia

vh2le@students.latrobe.edu.au; f.liu@latrobe.edu.au

² Faculty of Information Technology

Hanoi University of Technology

Hanoi, Vietnam

khangtd@it-hut.edu.au

Abstract. This paper presents fuzzy linguistic logic programming which is developed based on fuzzy logic programming introduced by P. Vojtáš in order to facilitate the representation and reasoning on knowledge expressed in natural language. In fuzzy linguistic logic programming, truth values are linguistic terms such as *true*, *very true*, *more or less true*, and *false* taken from a hedge algebra of the truth variable, and linguistic hedges, e.g., *very*, *more or less*, *quite*, and *rather*, which are frequently used in natural language, can be used as unary connectives in formulae. In order to compute the truth value of a query, we provide a computational model which directly manipulates linguistic terms. The soundness of the computational model is proved. A fixpoint semantics of logic programs and the completeness of the computational model are discussed.

Keywords: fuzzy logic programming; hedge algebra; linguistic value; linguistic hedge.

1 Introduction

People use words in natural language, which are inherently imprecise, vague and qualitative in nature, to describe real world information, to analyse, to reason, and to make decisions. Moreover, linguistic hedges (modifiers) are very often used to state different levels of emphasis. Therefore, it is necessary to investigate logical systems that can directly work with words and make use of linguistic hedges since such systems will make it easier to represent and reason on knowledge expressed in natural language.

Fuzzy logic which is derived from fuzzy sets theory, introduced by L. Zadeh, deals with reasoning that is approximate rather than exact. In fuzzy logic, the truth value domain is not the classical set $\{False, True\}$ or $\{0, 1\}$, but a set of linguistic truth values or the whole unit interval $[0,1]$. In fuzzy logic, linguistic hedges play an essential role in the generation of the values of a linguistic variable as well as in the modification of fuzzy predicates [13]. Fuzzy logic provides us with a very powerful tool towards handling imprecision and uncertainty, which are

very often encountered in real world information, and a capacity for representing and reasoning on knowledge expressed in linguistic forms.

Fuzzy logic programming, introduced in [11], is a formal model of an extension of logic programming without negation working with a truth functional fuzzy logic in narrow sense. In fuzzy logic programming, atoms and rules, which are many-valued implications, are graded to a certain degree in $[0,1]$, and a wide variety of many-valued connectives are allowed in formulae. A sound and complete computational model is provided to compute the truth value of a query.

Hedge algebras, introduced in [9] by C.H. Nguyen *et al.*, give an algebraic characterization of linguistic hedges. Hedge algebras have been shown to have a rich enough algebraic structure to represent linguistic domains, and based on them, several methods of linguistic reasoning, which directly manipulate linguistic terms, are developed [7,4]. Hedge algebras can also be effectively applied to fuzzy control, and in those methods, there is no need to manipulate any sort of membership functions at all [7,8].

We proceed as follows. Section 2 presents linguistic truth domains generated from hedge algebras, inverse mappings of hedges, many-valued modus ponens and connectives wrt linguistic truth domains while Section 3 presents fuzzy linguistic logic programming, defines declarative and procedural semantics, proves the soundness of the computational model, and discusses a fixpoint semantics and the completeness of the computational model. Section 4 concludes the paper.

2 Linguistic Truth Domains

2.1 Hedge Algebras

Linguistic truth values such as *True*, *Very True*, *Very More-or-less False*, and so on, can be considered to be generated from a set of generators (primary terms) $G = \{False, True\}$ by using hedges from a set $H = \{Very, More-or-less, \dots\}$ as unary operations. There exists a natural ordering among these values, with $a \leq b$ meaning that a indicates a degree of truth less than or equal to b , e.g., $True < VeryTrue$ and $False < More-or-less False$ ($a < b$ iff $a \leq b$ and $a \neq b$). Hence, this domain is a partially ordered set and can be described as an abstract algebra $AX = (X, G, H, \leq)$, where X is a terms-domain, G is a set of primary terms, H is a set of hedges, and \leq is a *semantics-based order relation* on X . Such algebras are called *hedge algebras* (HA) [9,10].

Hedges either increase or decrease the meaning of terms they modify, i.e., $\forall h \in H$ and $\forall x \in X$ either $hx \geq x$ or $hx \leq x$. Given two hedges h and k , it is said that: (i) h and k are *converse* if $\forall x \in X (hx \leq x$ iff $kx \geq x)$; (ii) h and k are *compatible* if $\forall x \in X (hx \leq x$ iff $kx \leq x)$; (iii) h modifies terms stronger or equal to k , denoted $h \geq k$, if $\forall x \in X (hx \leq kx \leq x)$ or $(x \leq kx \leq hx)$; (iv) h is *positive* wrt k , i.e., h strengthens the degree of modification of k , if $\forall x \in X (h k x \leq k x \leq x)$ or $(x \leq k x \leq h k x)$; (v) h is *negative* wrt k , i.e., h weakens the degree of modification of k , if $\forall x \in X (k x \leq h k x \leq x)$ or $(x \leq h k x \leq k x)$.

For each $x \in X$, the set of all terms generated from x is denoted by $H(x) = \{\sigma x \mid \sigma \in H^*\}$, where H^* is the set of all strings of symbols in H . There is an

important semantic property of hedges called *semantics heredity*, i.e., hedges modify the meaning of a term a bit, but mainly preserve its original meaning. Therefore, if $hx \leq kx$, then $H(hx) \leq H(kx)$.

Given a term $x = h_n \dots h_1 u$, there exists an index i such that the suffix $h_i \dots h_1 u$ of x is called a *canonical representation* of x wrt u , i.e., $x = h_i h_{i-1} \dots h_1 u$ and $h_i h_{i-1} \dots h_1 u \neq h_{i-1} \dots h_1 u$ and $h_j x = x$, for all $j > i$. It can be shown that each term has one and only one canonical representation. In this paper, all expressions of terms are assumed to be in the form of canonical representation.

Since we allow hedges to be unary connectives in formulae, there is a need to be able to compute the truth value of a hedge-modified formula from that of the original. To do that the notion of an *inverse mapping* of a hedge introduced in [1] is used. We will show later that in some cases an inverse mapping of a hedge is undeterminable if HAs under consideration are non-linear, i.e., there are compatible hedges that are incomparable. Therefore, in the sequel, we only deal with linear HAs. This is also in line with Vojtáš’s work. For a formal presentation of general HAs, we refer the reader to [9,10,6].

2.2 Linear Symmetric Hedge Algebras

The set G of generators usually has two comparable primary terms. HAs that have exactly two primary terms are called *symmetric* ones. For the truth variable, G consists of $False < True$, denoted c^- and c^+ , respectively. The set of hedges H can be divided into two disjoint subsets: $H^+ = \{h | hc^+ > c^+\} = \{h | hc^- < c^-\}$ and $H^- = \{h | hc^+ < c^+\} = \{h | hc^- > c^-\}$. Clearly, any pair of hedges in each subset H^+ or H^- are compatible, and every hedge in H^+ is converse to every hedge in H^- and vice versa. A HA $AX = (X, \{c^-, c^+\}, H, \leq)$ is called a *linear symmetric* HA if H^+ and H^- are linearly ordered. It can be shown that a terms-domain X generated from a linear symmetric HA is a linearly ordered set. From now on, if not stated otherwise, by “HA” we mean “linear symmetric HA”.

Let $I \notin H$ be the *identity* hedge, i.e., $\forall x \in X, Ix = x$. For a HA, it can be shown that I is the least element of both sets $H^+ \cup \{I\}$ and $H^- \cup \{I\}$.

Example 1. Consider a HA $AX = (X, \{c^-, c^+\}, \{V, M, P, Mol\}, \leq)$, where V, M, P , and Mol stand for *Very, More, Probably*, and *More-or-less*, respectively. AX is a linear symmetric HA as follows. H is decomposed into $H^+ = \{V, M\}$ and $H^- = \{P, Mol\}$. In $H^+ \cup \{I\}$, we have $I < M < V$ while in $H^- \cup \{I\}$, we have $I < P < Mol$.

It is shown in [10] that if $H \neq \emptyset$, and $H(c^+), H(c^-)$ are infinite, we have $inf(c^+) = sup(c^-)$, $sup(c^+) = 1$, and $inf(c^-) = 0$. Let $W = inf(c^+) = sup(c^-)$, thus $0 < c^- < W < c^+ < 1$. The *linguistic truth domain* from a HA $AX = (X, \{c^-, c^+\}, H, \leq)$ is defined as $\overline{X} = X \cup \{0, W, 1\}$, where $0, W, 1$ are respectively the least, the neutral and the greatest elements of the domain.

The usual *negation, conjunction, and disjunction* are defined as follows. Given $x = \sigma c$, where $\sigma \in H^*$ and $c \in \{c^+, c^-\}$, y is the negation of x , denoted $y = -x$, if $y = \sigma c'$ and $\{c, c'\} = \{c^+, c^-\}$, and $-0 = 1, -1 = 0, -W = W$. Let $x, y \in \overline{X}$, $x \wedge y = \min(x, y)$ and $x \vee y = \max(x, y)$.

Furthermore, we restrict ourselves to a class of HAs called *monotonic* HAs in order to define the notion of an inverse mapping of a hedge.

2.3 Monotonic Hedge Algebras

A HA $AX = (X, G, H, \leq)$ is called a *monotonic* HA if each $h \in H^+(H^-)$ is positive wrt all $k \in H^+(H^-)$ and negative wrt all $k \in H^-(H^+)$. For monotonic HAs, an *extended order relation* on $H \cup \{I\}$, denoted \leq_h , is defined based on the order relations on $H^+ \cup \{I\}$ and $H^- \cup \{I\}$ as follows. Given $h, k \in H \cup \{I\}$, $h \leq_h k$ iff: (i) $h \in H^-, k \in H^+$; or (ii) $h, k \in H^+ \cup \{I\}$ and $h \leq k$; or (iii) $h, k \in H^- \cup \{I\}$ and $h \geq k$. We denote $h <_h k$ if $h \leq_h k$ and $h \neq k$.

Example 2. The HA in Example 1 is also monotonic since V and M are positive wrt V and M and negative wrt P and Mol , and P and Mol are positive wrt P and Mol and negative wrt V and M . In addition, $Mol <_h P <_h I <_h M <_h V$.

The following proposition shows how to compare terms in a monotonic HA.

Proposition 1 ([1])

Given a monotonic HA $AX = (X, \{c^+, c^-\}, H, \leq)$. Let $h, k \in H$; $c_1, c_2 \in \{c^+, c^-\}$; $\sigma_1, \sigma_2, \delta \in H^*$; $x, y \in \overline{X}$. We have: (i) $x \leq y \Leftrightarrow -x \geq -y$; (ii) $h \leq_h k \Leftrightarrow h\sigma c^+ \leq k\sigma c^+$; (iii) $h \in H^- \Rightarrow h\sigma c^+ \leq \sigma c^+$; (iv) $h \in H^+ \Rightarrow h\sigma c^+ \geq \sigma c^+$; (v) $\sigma_1 c_1 \leq \sigma_2 c_2 \Leftrightarrow \sigma_1 h c_1 \leq \sigma_2 h c_2$; (vi) $\sigma_1 c_1 \leq \sigma_2 c_2 \Leftrightarrow \sigma_1 \delta c_1 \leq \sigma_2 \delta c_2$.

2.4 Inverse Mappings of Hedges

In fuzzy logic, knowledge is usually represented in terms of pairs of a *vague sentence* and its *degree of truth*, which is also expressed in linguistic terms. A vague sentence can be represented by $p(x, u)$, where x is a variable or a constant, u is a fuzzy concept, and $p(\cdot, \cdot)$ is a linguistic analog of the classical predicate. For example, the assertion “*It is quite true that John is studying hard*” can be represented by a pair $(study(John, hard), QuiteTrue)$.

According to [12], assessments “*It is true that John is studying very hard*” and “*It is very true that John is studying hard*” can be considered to be equivalent. Thus, hedge “*Very*” can be moved from the fuzzy concept to the truth value and conversely, and this is formalised in [7] by rules of moving hedges:

$$\begin{aligned} (RT1) \quad & (p(x, hu), \sigma c) \Rightarrow (p(x, u), \sigma hc) \\ (RT2) \quad & (p(x, u), \sigma hc) \Rightarrow (p(x, hu), \sigma c) \end{aligned}$$

However, in some cases none of the rules is applicable. For instance, given “*Very True*” being the truth value of “*John is young*”, we can not compute the truth value of “*John is probably young*”. The notion of an inverse mapping of a hedge provides us a solution to the problem.

Definition 1 (An inverse mapping of a hedge). Consider a monotonic HA $AX = (X, \{c^+, c^-\}, H, \leq)$ and a hedge $h \in H$. A mapping $h^- : \overline{X} \rightarrow \overline{X}$ is called an inverse mapping of h iff it satisfies the following conditions: (i) $h^-(\sigma hc) = \sigma c$, where $c \in \{c^+, c^-\}$, $\sigma \in H^*$; (ii) $x \leq y \Rightarrow h^-(x) \leq h^-(y)$.

The meaning of an inverse mapping of a hedge is that if we modify the fuzzy concept of a vague sentence by a hedge, its truth value will be changed by the inverse mapping of that hedge. Since $0, W, 1$ are fixed points¹, we have $\forall h \in H, h^-(0) = 0, h^-(W) = W, h^-(1) = 1$.

Condition (i) assures Rule *RT2* to be fulfilled, and intuitively an inverse mapping of a hedge should be monotonic (Condition (ii)). However, with non-monotonic HAs, we may not have an inverse mapping of a hedge that satisfies the two conditions simultaneously. For example, consider a HA containing two hedges *Very* $\in H^+$ and *Little* $\in H^-$. *Very* is positive wrt *Little*, e.g., *VeryLittleTrue* $<$ *LittleTrue* $<$ *True*, so the HA is not a monotonic HA. By Condition (i), we have *Little*⁻(*VeryLittleTrue*) = *VeryTrue* $>$ *True* = *Little*⁻(*LittleTrue*), i.e., *Little*⁻ is not monotonic.

Also, with non-linear HAs, in some cases we can not determine inverse mappings of hedges. For example, consider a HA having two incomparable hedges, say *P(robably), A(pproximately)* $\in H^-$. We can see that since *Ac*⁺ and *Pc*⁺ are incomparable, *P*⁻(*Ac*⁺) and *P*⁻(*Pc*⁺) = *c*⁺ should be incomparable too. This is unreasonable since every truth value is comparable to *c*⁺ and *c*⁻.

An inverse mapping of a string of hedges is defined as: $(h_p \dots h_1)^-(\delta c) = h_p^-(\dots(h_1^-(\delta c))\dots)$. Thus, sentence $p(x, h_p \dots h_1 u)$ can be interpreted as $p(x, (h_p \dots h_1 u))$, i.e., the string of hedges is considered as a whole modifier, or as $p(x, (h_p \dots (h_1 u) \dots))$, i.e., the fuzzy concept is modified by hedges one after the other, with the same truth value.

2.5 K-Limited Monotonic Hedge Algebras

In practice, we normally use finitely many degrees of quality or quantity to describe real world applications which are granulated (L. Zadeh). Also, it is reasonable to consider only finitely many truth values in order to provide a logical system that can be implemented for computers. In fact, we later show that with a finite truth domain we can obtain the Least Herbrand model for a finite program after a finite number of steps. Therefore, we can define a *K-limited monotonic HA*, where *K* is a positive integer, as a monotonic HA in which each truth value has at most *K* hedges, i.e., the length of the canonical representations of all terms wrt $c \in G$ is at most *K* + 1. For *K*-limited monotonic HAs, the definition of an inverse mapping of a hedge remains unchanged.

Example 3. Consider a 2-limited monotonic HA $AX = (X, \{c^+, c^-\}, H, \leq)$, where $H^+ = \{V, M\}, H^- = \{P, Mol\}$ with $Mol <_h P <_h I <_h M <_h V$. The following is an example of inverse mappings of hedges for the HA:

$$V^-(h_2 h_1 c^+) = \begin{cases} h_2 c^+ & \text{if } h_1 = V \\ P M o l c^+ & \text{if } h_1 = M, h_2 \in H^+ \cup \{I\} \\ h_2 M o l c^+ & \text{if } h_1 = M, h_2 \in H^- \\ M o l M o l c^+ & \text{otherwise} \end{cases}$$

¹ *x* is said to be a *fixed point* if $\forall h \in H, h x = x$ [10].

$$\begin{aligned}
 M^-(h_2h_1c^+) &= \begin{cases} h_2Vc^+ & \text{if } h_1 = V, h_2 \in H^+ \\ MVc^+ & \text{if } h_1 = V, h_2 \in H^- \cup \{I\} \\ h_2c^+ & \text{if } h_1 = M \\ PMolc^+ & \text{if } h_1 = I \\ PMolc^+ & \text{if } h_1 = P, h_2 \in H^+ \cup \{I\} \\ h_2Molc^+ & \text{if } h_1 = P, h_2 \in H^- \\ MolMolc^+ & \text{otherwise} \end{cases} \\
 P^-(h_2h_1c^+) &= \begin{cases} h_2Vc^+ & \text{if } h_1 = V, h_2 \in H^+ \\ MVc^+ & \text{if } h_1 = V, h_2 \in H^- \cup \{I\} \\ MVc^+ & \text{if } h_1 = M \text{ or } h_1 = I \\ h_2c^+ & \text{if } h_1 = P \\ PMolc^+ & \text{if } h_1 = Mol, h_2 \in H^+ \cup \{I\} \\ h_2Molc^+ & \text{if } h_1 = Mol, h_2 \in H^- \end{cases} \\
 Mol^-(h_2h_1c^+) &= \begin{cases} VVc^+ & \text{if } h_1 \in \{V, M, I\} \\ h_2Vc^+ & \text{if } h_1 = P, h_2 \in H^+ \\ MVc^+ & \text{if } h_1 = P, h_2 \in H^- \cup \{I\} \\ h_2c^+ & \text{if } h_1 = Mol \end{cases}
 \end{aligned}$$

and $k^-(h_2h_1c^-) = -k^-(h_2h_1c^+)$, obtained by replacing c^+ by c^- . As usual, $\forall h \in H, h^-(0) = 0, h^-(W) = W, h^-(1) = 1$. It can be easily verified that these mappings satisfy the conditions (i) and (ii).

We can see that an inverse mapping of a hedge is not unique since we can build many that satisfy the conditions. For the above example, we can define:

$$V^-(h_2h_1c^+) = \begin{cases} h_2c^+ & \text{if } h_1 = V \\ MolMolc^+ & \text{otherwise.} \end{cases}$$

This is acceptable since reasoning based on fuzzy logic is approximate, and thus inverse mappings of hedges should be built according to applications.

2.6 Many-Valued Modus Ponens

Our logic is truth-functional, i.e., the truth value of a compound formula, built from its components using a logical connective, is a function, which is called the truth function of the connective, of the truth values of the components.

The computational model is developed based on many-valued modus ponens. To guarantee the soundness of modus ponens, the truth function of the implication (called *implicator*) must be *residual* to the t-norm evaluating modus ponens (see [2]). Precisely, let r be the truth value of an implication $h \leftarrow b$, \mathcal{C} a t-norm, and \leftarrow^\bullet its residual implicator; we have:

$$\mathcal{C}(b, r) \leq h \text{ iff } r \leq \leftarrow^\bullet (h, b).$$

Recall that in many-valued logics there are several prominent sets of connectives called Łukasiewicz and Gödel ones. Given a linguistic truth domain \overline{X} , since all

the values are linearly ordered, we assume that they are $v_0 \leq v_1 \leq \dots \leq v_n$, where $v_0 = 0, v_n = 1$. The Łukasiewicz t-norm and implicator can be defined as:

$$\begin{aligned} \mathcal{C}_L(v_i, v_j) &= \begin{cases} v_{i+j-n} & \text{if } i + j - n > 0 \\ v_0 & \text{otherwise} \end{cases} \\ \leftarrow_{\bullet L}(v_j, v_i) &= \begin{cases} v_n & \text{if } i \leq j \\ v_{n+j-i} & \text{otherwise} \end{cases} \end{aligned}$$

and those of Gödel can be:

$$\begin{aligned} \mathcal{C}_G(v_i, v_j) &= \min(v_i, v_j) \\ \leftarrow_{\bullet G}(v_j, v_i) &= \begin{cases} v_n & \text{if } i \leq j \\ v_j & \text{otherwise} \end{cases} \end{aligned}$$

We can see that t-norms are monotonic in all arguments, and implicators are non-decreasing in the first argument and non-increasing in the second.

3 Fuzzy Linguistic Logic Programming

3.1 Language

Like [11], our language is a many sorted (typed) predicate language. Let \mathcal{A} denote the set of all attributes. For each sort of variables $A \in \mathcal{A}$, there is a set \mathcal{C}^A of constant symbols, which are names of elements of the domain of A . In order to achieve the Least Herbrand model after a finite number of steps, we do not allow any function symbols. Connectives can be: conjunctions \wedge (also called Gödel conjunction with the truth function being Gödel t-norm), \wedge_L (Łukasiewicz conjunction with the truth function being Łukasiewicz t-norm); the disjunction \vee ; implications \leftarrow_L (Łukasiewicz), \leftarrow_G (Gödel); and hedges as unary connectives. For a connective c different from hedges, its truth function is denoted by c^\bullet , and for a hedge connective h , its truth function is its inverse mapping h^- .

A *term* is either a constant or a variable. An *atom* is of the form $p(t_1, \dots, t_n)$, where p is an n-ary predicate symbol, and t_1, \dots, t_n are terms of corresponding attributes. A *body formula* is defined inductively as follows: (i) An atom is a body formula. (ii) If F and G are body formulae, then so are $\wedge(F, G), \vee(F, G)$ and hF , where h is a hedge (here we use the prefix notation for connectives in body formulae). A *rule* is a graded implication $(A \leftarrow B.r)$, where A is an atom called *rule head*, B is a body formula called *rule body*, and r is a truth value different from 0. $(A \leftarrow B)$ is called the *logical part* of the rule. A *fact* is a graded atom $(A.b)$, where A is an atom called the logical part of the fact, and b is a truth value different from 0. A *query* is an atom used as a question $?A$.

Definition 2 (Fuzzy linguistic logic program). A fuzzy linguistic logic program (*program, for short*) is a finite set of rules and facts, where truth values are from the linguistic truth domain of a K -limited monotonic HA, hedges used in body formulae (if any) belong to the set of hedges of the HA, and there are no two rules (facts) having the same logical part, but different truth values.

We assume as usual that the underlying language of a program P is defined by constants (if no such constant exists, we add some constant such as a to form ground terms) and predicate symbols appearing in P , so we can now refer to *Herbrand universe* of a sort A , which consists of all ground terms of A , by U_P^A and *Herbrand base* of P , which consists of all ground atoms, by B_P (see [5]).

A program P can be represented as a partial mapping:

$$P : \text{Formulae} \rightarrow \overline{X} \setminus \{0\}$$

where the domain of P , $dom(P)$, is finite and consists only of logical parts of rules and facts, and \overline{X} is a linguistic truth domain. The truth value of a rule ($A \leftarrow B.r$) is $r = P(A \leftarrow B)$, and that of a fact ($A.b$) is $b = P(A)$.

Since in our logical system we just want to obtain computed answers for queries, we do not look for 1-tautologies to extend the capabilities of the system although we can have some due to the fact that our connectives are classical many-valued ones (see [2]).

3.2 Declarative Semantics

Since we are working with logic programs without negation, it is reasonable to consider only fuzzy Herbrand interpretations and models. Given a program P , let \overline{X} be the linguistic truth domain; a *fuzzy linguistic Herbrand interpretation* (interpretation, for short) f is a mapping $f : B_P \rightarrow \overline{X}$. The ordering \leq in \overline{X} can be extended to the set of interpretations as follows. We say $f_1 \sqsubseteq f_2$ iff $f_1(A) \leq f_2(A)$ for all ground atoms A . Clearly, the set of all interpretations of a program is a complete lattice under \sqsubseteq . The least interpretation called the *bottom interpretation* maps every ground atom to 0.

An interpretation f can be extended to all ground formulae, denoted \overline{f} , by using the unique homomorphic extension as follows: (i) $\overline{f}(A) = f(A)$, if A is a ground atom; (ii) $\overline{f}(c(B_1, B_2)) = c^\bullet(\overline{f}(B_1), \overline{f}(B_2))$, where B_1, B_2 are ground formulae, and c is one of the binary connectives; (iii) $\overline{f}(hB) = h^-(\overline{f}(B))$, where B is a ground body formula, and h is a hedge. For non-ground formulae, since all the formulae in our language are considered universally quantified, \overline{f} is defined as $\overline{f}(\varphi) = \overline{f}(\forall\varphi) = \text{inf}_\vartheta\{\overline{f}(\varphi\vartheta) \mid \varphi\vartheta \text{ is a ground instance of } \varphi\}$, where $\forall\varphi$ means universal quantification of all variables with free occurrence in φ .

An interpretation f is a *model* of a program P if for all $\varphi \in dom(P)$, $\overline{f}(\varphi) \geq P(\varphi)$. Therefore, $P(\varphi)$ is understood as a lower bound for the truth value of φ .

Definition 3 (Correct answer). *Given a program P , let \overline{X} be the linguistic truth domain. A pair $(x; \theta)$, where $x \in \overline{X}$, and θ is a substitution, is called a correct answer for P and $?A$ if for any model f of P , $\overline{f}(A\theta) \geq x$.*

3.3 Procedural Semantics

Given a program P and a query $?A$, we want to compute a lower bound for the truth value of A under any model of P . Recall that in the theory of many-valued modus ponens [2], given $(A \leftarrow_i B.r)$ and $(B.b)$, we have $(A.C_i(b, r))$, where C_i is

the t-norm evaluating the modus ponens. Like in [11], our computational model utilises the so-called *admissible rules*.

Definition 4 (Admissible rules). Admissible rules are defined as follows:

Rule 1. From $((XA_mY); \vartheta)$ infer $((XC(B, r)Y)\theta; \vartheta\theta)$ if: (i) A_m is an atom (called the selected atom); (ii) θ is an mgu of A_m and A ; (iii) $(A \leftarrow B.r)$ is a rule in the program.

Rule 2. From (XA_mY) infer $(X0Y)$ (this rule is used for situations where A_m does not unify with any rule head or logical part of facts in the program).

Rule 3. (for hedges) From $(XhBY)$ infer $(Xh^-(B)Y)$ if B is a non-empty body formula.

Rule 4. From $((XA_mY); \vartheta)$ infer $((XrY)\theta; \vartheta\theta)$ if: (i) A_m is an atom; (ii) θ is an mgu of A_m and A ; (iii) $(A.r)$ is a fact in the program.

Rule 5. If there are no more predicate symbols in the word, replace all connectives \wedge 's and \vee 's with \wedge^\bullet and \vee^\bullet , respectively. Then, since this word contains only some additional C 's, h^- 's and truth values, evaluate it. The substitution remains unchanged.

Note that the above rules except Rule 3 are the same as those in [11].

Definition 5 (Computed answer). Let P be a program and $?A$ a query. A pair $(r; \theta)$, where r is a truth value, and θ is a substitution, is said to be a computed answer if there is a sequence G_0, \dots, G_n such that: (i) $G_0 = (A; id)$; (ii) Every G_{i+1} is inferred from G_i by one of admissible rules (here we also utilise the usual Prolog renaming of variables along derivation); (iii) $G_n = (r; \theta')$ and $\theta = \theta'$ restricted to variables of A . The computation is said to have length n .

Let us give an example of computations.

Example 4. Assume that our truth domain is from the 2-limited monotonic HA described in Example 3 $AX = (X, \{False, True\}, \{V, M, P, Mol\}, \leq)$, and we have a knowledge base consisting of the following statements: (i) Sentence “If a student studies very hard, and his/her university is more-or-less high-ranking, then he/she will be a good employee” has a degree of truth “Very Probably True”; (ii) Sentence “The university where Ann is studying is probably high-ranking” has a degree of truth “True”; (iii) Sentence “Ann is studying hard” has a degree of truth “More Very True”.

Let $GdEm$, $StHd$, $HiraUn$, and T stand for “good employee”, “study hard”, “high-ranking university”, and “True”, respectively. Then, the knowledge base can be represented by the following program:

$$\begin{aligned} &(GdEm(x) \leftarrow \wedge(VStHd(x), MolHiraUn(x)).VPT) \\ &(HiraUn(Ann).PT) \text{ (by } RT1, (PHiraUn(Ann).T) \Rightarrow (HiraUn(Ann).PT)) \\ &(StHd(Ann).MVT). \end{aligned}$$

Given a query $?GdEm(Ann)$, we can have the following computation (since the query is ground, the substitution in the computed answer is the identity):

$$?GdEm(Ann)$$

$$\begin{aligned}
 & \mathcal{C}(\wedge(VStHd(Ann), MolHiraUn(Ann)), VPT) \\
 & \mathcal{C}(\wedge(V^-(StHd(Ann)), MolHiraUn(Ann)), VPT) \\
 & \mathcal{C}(\wedge(V^-(StHd(Ann)), Mol^-(HiraUn(Ann))), VPT) \\
 & \mathcal{C}(\wedge(V^-(MVT), Mol^-(HiraUn(Ann))), VPT) \\
 & \mathcal{C}(\wedge(V^-(MVT), Mol^-(PT)), VPT) \\
 & \mathcal{C}(\wedge^\bullet(V^-(MVT), Mol^-(PT)), VPT)
 \end{aligned}$$

If we use the inverse mappings of hedges built in Example 3, Łukasiewicz t-norm, and take *min* as the truth function of the conjunction, we have: $\mathcal{C}(\wedge^\bullet(V^-(MVT), Mol^-(PT)), VPT) = \mathcal{C}(min(MT, MVT), VPT) = \mathcal{C}(MT, VPT) = PMolT$. Hence, the sentence “*Ann will be a good employee*” is at least “*Probably More-or-less True*”. Note that if we use Gödel t-norm, we will have $(GdEm(Ann).VPT)$.

3.4 Soundness of the Computational Model

Theorem 1 (Soundness of the computational model). *Every computed answer for a program P and a query ?A is a correct answer.*

Proof. Since inverse mappings of hedges are monotonic, the proof is similar to that in [11].

3.5 Fixpoint Semantics and Completeness

An immediate consequence operator can be defined as follows:

Definition 6 (Immediate consequence operator). *Let P be a program. The operator T_P mapping from interpretations to interpretations is defined as follows. For every interpretation f and every ground atom $A \in B_P$,*

$$T_P(f)(A) = \max\{\sup\{C_i(\bar{f}(B), r) : (A \leftarrow_i B.r) \text{ is a ground instance of a rule in } P\}, \sup\{b : (A.b) \text{ is a ground instance of a fact in } P\}\}.$$

It is shown in [3] that T_P is continuous, and furthermore an interpretation *f* is a model of a program *P* iff $T_P(f) \sqsubseteq f$. Therefore, the Least Herbrand model of the program *P* is exactly the least fixpoint of T_P and can be obtained by iterating T_P from the bottom interpretation after ω iterations, where ω is the smallest limit ordinal (apart from 0). Moreover, since the truth domain \bar{X} and Herbrand base B_P are finite, the least model of the program *P* can be obtained after at most $\mathcal{O}(|P||\bar{X}|)$ steps, where $|S|$ denotes the cardinality of a set *S*. This is an important tool for dealing with recursive programs, for which computations can be infinite.

Theorem 2 (Completeness of the computational model). *Let P be a program and ?A a query. For every correct answer $(x; \theta)$ for P and ?A, there exists a computed answer $(r; \sigma)$ for P and ?A and a substitution γ such that $r \geq x$ and $\theta = \sigma\gamma$.*

The theorem can be proved based on the fixpoint semantics and extended versions of Mgu and Lifting lemmas [5] as shown in [3].

4 Conclusion

We have presented fuzzy linguistic logic programming as a result of integrating hedge algebras and fuzzy logic programming. The main purpose of this work is to facilitate the representation and reasoning on knowledge stated in natural language, in which vague sentences are often given a degree of truth expressed in linguistic terms rather than a number, and linguistic hedges are usually used to indicate different levels of emphasis. In a fuzzy linguistic logic program, each fact or rule is graded to a certain degree specified by a truth value from the linguistic truth domain of a hedge algebra of the truth variable, and hedges, which are not allowed in fuzzy logic programming, can be used as unary connectives in body formulae. The allowance of linguistic hedges as unary connectives strictly makes the expressive power of the language increase. To compute the truth value of a query, we provide a computational model which directly computes with linguistic terms. It has been shown that a knowledge base expressed in natural language can be easily represented in our language. The soundness of the computational model is proved. A fixpoint semantics for logic programs and the completeness of the computational model are discussed.

References

1. Dinh-Khac, D., Hölldobler, S., Tran, D.K.: The Fuzzy Linguistic Description Logic ALC_{FL} . In: 11th Intern. Conf. on Info. Processing and Man. of Uncertainty in Knowl.-Based Syst. (IPMU 2006), Paris, pp. 2096–2103 (2006)
2. Hájek, P.: *Metamathematics of Fuzzy Logic*. Kluwer Academic Publishers, Dordrecht (1998)
3. Le, V.H., Liu, F., Tran, D.K.: Fixpoint Semantics and Completeness of the Computational Model for Fuzzy Linguistic Logic Programming. In: Huang, D.S., et al. (eds.) ICIC 2008. LNCS (LNAI) (to appear, 2008)
4. Le, V.H., Tran, D.K., Dinh-Khac, D.: Inverse Mappings of Hedges and their Application to Fuzzy Reasoning. In: 3rd National Symp. on Fundamental and Applied Info. Tech. Research (FAIR 2007) (in Vietnamese) (2007)
5. Lloyd, J.W.: *Foundations of Logic Programming*. Springer, Berlin (1987)
6. Nguyen, C.H.: A Topological Completion of Refined Hedge Algebras and a Model of Fuzziness of Linguistic Terms and Hedges. *Fuzzy Sets and Syst.* 158, 436–451 (2007)
7. Nguyen, C.H., Tran, D.K., Huynh, V.N., Nguyen, H.C.: Hedge Algebras, Linguistic-Valued Logic and their Application to Fuzzy Reasoning. *Intern. J. of Uncertainty, Fuzziness and Knowl.-Based Syst.* 7, 347–361 (1999)
8. Nguyen, C.H., Vu, N.L., Le, X.V.: Optimal Hedge-Algebras-Based Controller: Design and Application. *Fuzzy Sets and Syst.* 159, 968–989 (2008)
9. Nguyen, C.H., Wechler, W.: Hedge Algebras: An Algebraic Approach to Structure of Sets of Linguistic Truth Values. *Fuzzy Sets and Syst.* 35, 281–293 (1990)
10. Nguyen, C.H., Wechler, W.: Extended Hedge Algebras and their Application to Fuzzy Logic. *Fuzzy Sets and Syst.* 52, 259–281 (1992)
11. Vojtáš, P.: Fuzzy Logic Programming. *Fuzzy Sets and Syst.*, 124, 361–370 (2001)
12. Zadeh, L.A.: The Concept of a Linguistic Variable and Its Application in Approximate Reasoning I, II, III. *Info. Sciences* 8, 199–249, 301–357, 9, 43–80 (1975)
13. Zadeh, L.A.: Knowledge Representation in Fuzzy Logic. *IEEE Transactions on Knowledge and Data Engineering* 1, 89–99 (1989)

Improve Flow Accuracy and Byte Accuracy in Network Traffic Classification

Haitao He^{1,3}, Chunhui Che¹, Feiteng Ma², Xiaonan Luo^{1,3}, and Jianmin Wang^{1,3}

¹Computer Application Institute, Sun Yat-Sen University, Guangzhou 510275, China
hthe@mail.sysu.edu.cn

²School of Information Science and Technology, Sun Yat-Sen University,
Guangzhou 510275, China
is03mft@mail2.sysu.edu.cn

³Key Laboratory of Digital Life (Sun Yat-sen University), Ministry of Education,
Guangzhou 510275, China
mcswj@mail.sysu.edu.cn

Abstract. Most of the current network traffic classification approaches employ single classifier method with achieving lower accuracy under small training set. Different from high flow accuracy, byte accuracy, as an important metric for network traffic classification, is usually ignored by many researchers. To address these two problems, this paper proposes a novel classification algorithm. It combines ensemble learning with cost-sensitive learning, which enables the classification model to achieve high flow accuracy as well as byte accuracy. By evaluating our algorithm with the real 7-day traces collected at the edge of the campus network, the results show that it can averagely obtain flow accuracy of 94% as well as byte accuracy of 81%.

Keywords: network traffic classification; ensemble learning; cost-sensitive learning; byte accuracy.

1 Introduction

At the edge of enterprise or campus networks, usually some policies would be applied to guarantee the SLA (Service Level Agreement) for some special applications (WEB, VoIP et al). This requires the network equipments having the ability to identify the application for IP flow. Accurate classification of network traffic is an essential step for many other topics, including security, network administration, and trouble shooting. Therefore, traffic classification attracts a lot of research interests in the past few years.

In early literatures, port-based identification approach was widely used in network traffic classification, as traditional applications use standard ports assigned by IANA (for instance, WEB traffic uses port 80 and SMTP uses port 25). However, since the year 2002, in order to break through firewalls and other network security equipments, an increasing number of network applications would no longer use the standard ports for communicating, while the dynamic port allocation and camouflage technology have been widely used. The effectiveness of port-based identification approach was

greatly undermined, and researches showed that it could not obtain more than 70% accuracy rate nowadays.

An alternative approach is payload-based identification. It identifies network traffic by searching the packet payload for signatures of known applications [1,2]. This approach is very useful and employed by many commercial bandwidth management products [3]. However, payload-based identification has several limitations. First, this technology can only identify network traffic whose signatures are known as prior. Second, it brings great challenge to the processing and storage capacity of network equipment, especially in high-bandwidth environment. Third, the payload encryption technology, the tunnel technology and the evolution of Internet applications have further reduced the effectiveness of payload-based approach.

Considering the limitations mentioned above, more and more researchers paid attention to machine learning approaches using statistical flow characterizations in the past few years. According to Erman.J[4], there are several reasons why these approaches are recommended. First, different applications have different behaviors and thus exhibit different flow statistics. For example, Peer-to-Peer applications would have larger average packet size while IM client would have the smaller one. Second, although obfuscation of flow statistics is possible, it is generally difficult to implement. Third, classification based on flow statistics can benefit from a lot of work on flow sampling/estimation techniques.

Recent machine approaches generally employ *single classifier* method. It classifies examples depending on one classifier's decision; therefore, it is difficult to achieve a high accuracy without a large amount of training data. Second, previous works are mainly focused on achieving high flow accuracy. However, the high byte accuracy is also important for a network traffic classification system, for that in Internet the majority of the IP flows are small in bytes ("mice" flows) and only a few number of large flows ("elephant" flows) hold up the majority of traffic[5]. In our own traffic classification experience, the top 15% largest flows account for 81% of the traffic in terms of bytes. One extreme situation is that although we may attain 93% flow accuracy, the bytes accuracy is only 60%. From this perspective, byte accuracy may play a much more important role than flow accuracy in network traffic classification. Note that, in this paper, flow accuracy refers to the number of correctly classified flows to the total number of flows in a trace and byte accuracy is the number of correctly classified bytes to the total number of bytes in the trace.

In this paper, a novel machine learning algorithm for traffic classification is proposed, which classifies network traffic according to application types (Web, FTP, IM, Game, etc.). To address the above problems, this algorithm employs two learning paradigms: *ensemble learning* and *cost-sensitive learning*. *Ensemble learning* is a learning paradigm that constructs a set of classifiers and then classifies new examples by taking a (weighted) vote of their predictions. By combining the decisions of a set of classifiers, *ensemble learning* can achieve a much better performance than the single classifier. *Cost-sensitive learning* uses the idea of cost sensitivity when measuring the performance of a traffic classifier, which makes the misclassification of "expensive" traffic more costly. *Cost-sensitive learning* can help to improve the bytes accuracy rate of classification while maintaining high flow accuracy. By combining *ensemble learning* with *cost-sensitive learning*, our algorithm can achieve high flow accuracy as well as high byte accuracy. We evaluate our algorithm with one week

traces captured at the edge of south campus of Sun Yat-Sen University in China, and the results show that *ensemble learning* and *cost-sensitive learning* do help to improve the performance of network traffic classification.

The remainder of this paper is structured as follows. Section 2 presents our classification model. Section 3 describes the dataset used in this work and our experimental methodology. Then experimental results and analysis will be presented in section 4. Finally, concluding remarks and ideas for future work end this paper.

2 Traffic Classification Model

Given a labeled example set $S = \{(x_1, y_1), (x_2, y_2), \dots, (x_{|S|}, y_{|S|})\}$, the goal of traffic classification is to define a function $y=f(x)$ such that each x_i is assigned to one predefined traffic class y_i . The x_i values are typically vectors of the form $\langle x_{i1}, x_{i2}, \dots, x_{in} \rangle$ with n features. The y values are typically drawn from a discrete set of classes $\{y_1, y_2, \dots, y_m\}$. $|S|$ denotes the size of labeled example set S . In this paper, we would further consider how to distinguish unknown application types from known ones.

Generally speaking, the accuracy rate achieved by the single classifier is limited. A learning algorithm can be viewed as searching a space η of hypotheses to identify the best hypothesis in the space. When the amount of training data available is too small compared to the size of the hypothesis space, the learning algorithm can not find the best hypotheses in η that closed to the real one. In this paper, a paradigm called *ensemble learning* is employed. By constructing an ensemble out of all of those classifiers, the algorithm can “average” their votes and reduce the risk of choosing the wrong answer. Moreover, several researches [6] show that ensemble technique can further improve the applicability of algorithm when classifying different data sets.

In this paper, we consider an ensemble strategy that is similar to Bagging [7]. For each component classifier h_i ($i = 1, 2, \dots, T$), a training set with size N is resampled from the original training data with *bootstrap sampling* strategy. Each training set has the same size as the original training data, but some examples may not appear in it while others appear more than once. Then the learning system generates T classifiers from these training sets. When classifying a newly example x , a vote for class k is recorded by every component classifier for which $h_i(x) = k$ and the ensemble result $H(x)$ is then the class with the *majority voting*. Considering that not all types of applications generating flows are known as *a priori* and new ones may appear over time, a simple strategy is employed to distinguish unknown application types from known ones. When T classifiers examine one example x and the number of classifiers voting for a particular class does not meet a preset threshold θ , we believe that this example belongs to an unknown application.

$$H(x) = \begin{cases} \arg \max \sum_{i=0}^N f(x) & \geq \theta \\ \text{unknown application} & < \theta \end{cases} \quad (1)$$

Most of the currently-available algorithms for classification are designed to minimize the *error rate* (the number of incorrect predictions made), which implicitly assumes that all errors are equally costly. As described before, byte accuracy may play a

more important role than flow accuracy in network traffic classification, which makes problem cost-sensitive. In this paper, *cost-sensitive learning* is employed to obtain a high flow accuracy rate as well as high bytes accuracy rate.

In general, misclassification costs can be described by an cost matrix C , which $C(i,j)$ indicates the cost of predicting example x to class i when in fact it belongs to class j . For traffic classification, the simplest way to measure the cost matrix C is according to the overall byte rate of each application type. Given an example x and the probability of each class j $P(j|x)$, then the optimal prediction for x is the class i that minimizes:

$$R(i|x) = \sum_j P(j|x)C(i,j) \quad (2)$$

The *conditional risk* $R(i|x)$ is the expected cost of predicting that x belongs to class i . The goal of *cost-sensitive learning* is to achieve the lowest possible overall cost.

Current *cost-sensitive learning* research falls into three categories. The first is concerned with making particular classifier learning cost-sensitive [8,9]. The second uses Bayes risk theory to assign each example to its lowest risk class [10,11], as formula (2) shows. The third category converts cost-sensitive learning problems to cost-insensitive learning problems [12].

Table 1. The Pseudo-code of Classification Model

<p>Input: <i>the training set S, the confidence Threshold θ, the cost matrix C, the number of classifier T</i></p> <p>Process:</p> <pre style="margin-left: 20px;"> /** construct a ensemble classifier */ for $i \in \{1, \dots, T\}$ do $S_i \leftarrow \text{BootstrapSample}(S)$ $h_i \leftarrow \text{BuildClassifier}(S_i)$ end for /**reassign class values to minimize expected cost*/ for each example x in S do for each class j do $P(j x) \leftarrow \frac{1}{\sum_i 1} \sum_i P(j x, h_i)$ end for x's class $\leftarrow \arg \min_i \sum_j P(j x)C(i,j)$ end for /** reconstruct a ensemble classifier */ for $i \in \{1, \dots, T\}$ do $S_i \leftarrow \text{BootstrapSample}(S)$ $h_i \leftarrow \text{BuildClassifier}(S_i)$ end for </pre>
<p>Output: <i>an ensemble classifier with T classifiers</i></p>

Individually making each classification algorithm cost-sensitive is laborious, and often non-trivial. To make our algorithm more universal, this paper employs a strategy that belongs to the last category. Specifically, we consider a strategy similar to MetaCost [12]. Before learning a classifier, m training sets are generated by bootstrap sampling from the original training set, and then m classifiers are learned from each. Each class's probability for each example is estimated by the fraction of votes received from the ensemble, and then we can use formula (2) to reassign each training example to the estimated optimal class. By this way, we can get a new training set which minimizes the overall expected cost, and we can learn a cost-sensitive classifier on it even if the learning algorithm is cost-insensitive. This procedure treats the classifier as a black box, requiring no knowledge of its functioning or change to it.

In this paper, a new classification approach is proposed, which extends *ensemble learning* by incorporating *cost-sensitive learning*. Pseudo-code for classification procedure is shown in Table 1.

3 Datasets

3.1 Data Collection

To facilitate our work, several Internet traffic traces are needed for training and evaluating our classification model. The traces used in our work come from the south campus network of Sun Yat-Sen University in China. The measurement equipment is deployed at the edge router which links to the WAN. Fig.1 shows the whole network measurement system used in this paper.

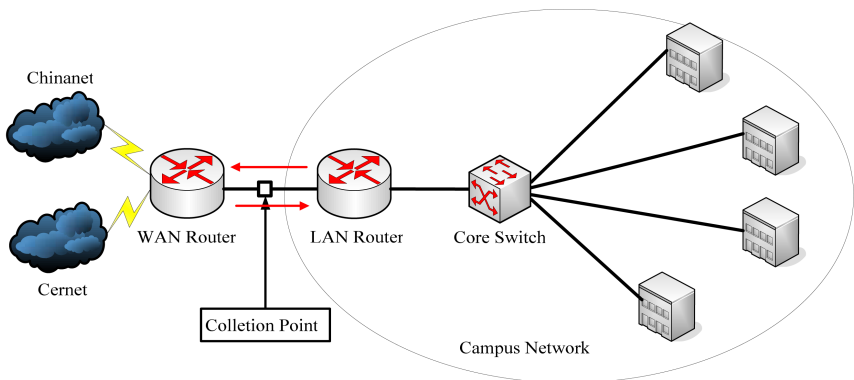


Fig. 1. Diagram of Network Measurement

Considering the *day-pattern* and *week-pattern* of network traffic, one week traces were collected. Specifically, we captured twenty-one traces between January 7-13, 2008, on 04:00 AM, 10:00 AM and 10:00 PM of each day. Because of the high-bandwidth of the edge of the campus network (>200Mbps) and the limited disk capacity of our network measurement system, only a five-minute-long bidirectional network traffic is captured each time. According to statistics, over 70% traffic volume

belongs to the student dormitory, accordingly we use the traffic coming to or from the male/female dormitories' to create and evaluate our classification model.

3.2 Application Types and Flow Characterized Statistics

To collect the statistical flow information necessary for the classification evaluations, the flows must be identified within the traces. In this paper, flow is defined as a bidirectional exchange of packets between two nodes. Note that, the flows within the traces we selected were all hand classified. Seven types of application are identified, which are *Web*, *PPLive*, *Xunlei*, *FTP*, *MSN*, *QQ* and *QQGame*, as Table 2 shows. Except these 7 types, the other applications which can not be hand classified are labeled as *Unknown*.

Table 2. Summary statistics of the datasets

Application Type	Flow		Byte (MB)	
	Amount	Ratio	Amount	Ratio
Web	920430	50.42%	21102.44	18.59%
PPLive	32481	1.78%	2267.30	2.00%
Xunlei	215892	11.83%	40406.86	35.59%
FTP	20698	1.13%	12.94	0.01%
MSN	3356	0.18%	10.19	0.01%
QQ	14887	0.82%	75.11	0.07%
QQGame	1002	0.05%	35.91	0.03%
Unknown	616681	33.78%	49621.60	43.71%
Total	1825427	100.00%	113532.35	100.00%

When defining flow characteristics, we consider flow features which are independent to packet payload and easy to compute. As a result, the following typical IP flow characteristics are taken into account.

- ◆ Flow volume in packets and bytes
- ◆ Flow duration and rate (bytes/duration)
- ◆ Upload download ratio (upload bytes/download bytes)
- ◆ Packet length (mean, standard deviation)
- ◆ Inter-arrival time between packets (mean, standard deviation)
- ◆ Distribution of packets (density [13], burstiness [14])

As each flow consists of two directions (upload and download), characteristics are calculated in both directions (except upload download ratio).

4 Experimental Results and Analysis

In order to evaluate the effectiveness of our classification model, three test scenarios are considered in this section: the comparison of flow accuracy rate using ensemble technique or not, the comparison of byte accuracy rate using cost-sensitive technique or not, and we would show the statistical information when applying our classification model to the actual network traffic classification work.

For each data set, ten-fold cross validation is employed for evaluation. In each fold, training data are randomly selected from each data set, with the same size 2000.

When classification model is built, the entire respective data set are used for evaluating. Because of space limited, we only report the average result about data sets on 10:00 AM.

C4.5 Decision Tree [15] and Random Tree [16] classifiers are used in the experiments, as the base classifier of ensemble technique. According to Zhou [17], a large size of ensemble does not necessarily lead to better performance of an ensemble, the ensemble size N in our classification model is not supposed to be too large. In the experiments, the value of N is set to 10. The confident threshold θ is set to 0.5, then a newly example is regarded as unknown application if the number of classifiers voting for a particular class does not exceed 5. To facilitate description, C4.5 Decision Tree and Random Tree are denoted by $C45$ and $RTree$ in the following discussion.

Fig.3 shows the flow accuracies achieved by each base classifier and its ensemble form. Overall, we find that ensemble technique does help to improve the flow accuracy rate of $C45$ and $TRree$. On average, the flow accuracy rate of $C45$ is improved from 89% to 93%, and the flow accuracy rate of $RTree$ is improved from 85% to 94%, respectively. By comparing Fig.2-a with Fig.2-b, we found that $C45$ obtained a higher flow accuracy than $RTree$ while obtaining a little bit lower ensemble flow accuracy. This is because $RTree$ can better maintain the diversity in the ensemble learning process, which ensures any two component classifiers can still be diverse even if their training data are similar.

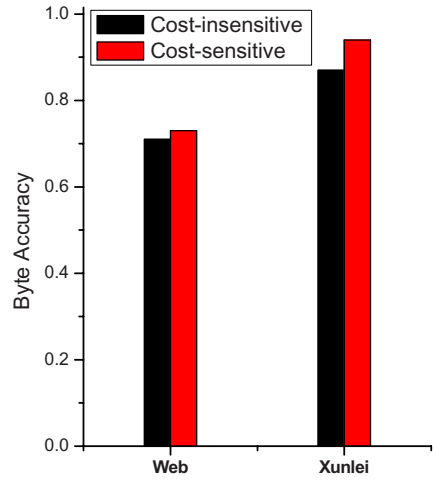
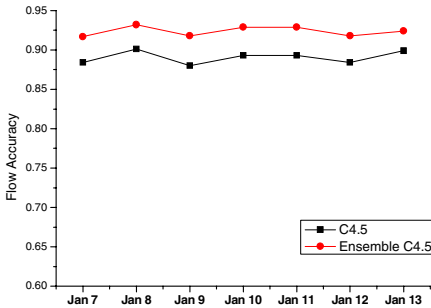
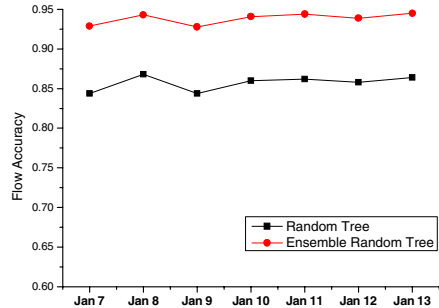


Fig. 1. Byte Accuracy by Application



a) C4.5 Decision Tree



b) Random Tree

Fig. 3. Flow Accuracy Comparison

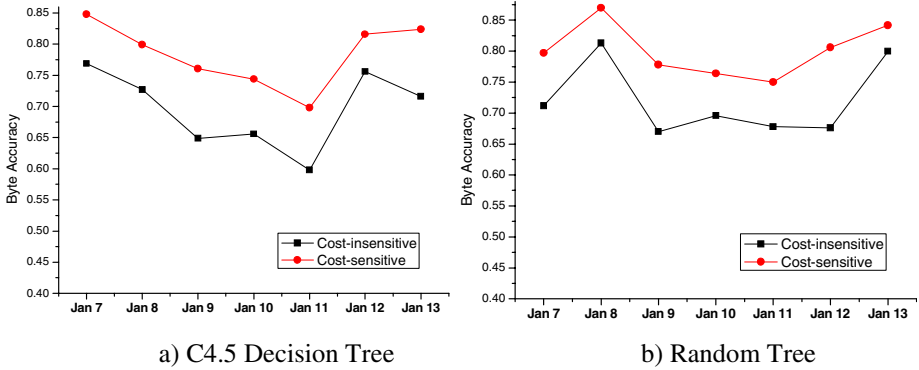


Fig. 4. Byte Accuracy Comparison

By introducing cost-sensitive technique into learning process, a higher byte accuracy rate can be achieved, as Fig.4 shows. For *C45*, an average byte accuracy of 80% can be obtained, comparing to the 69% without *cost-sensitive learning*. For *RTree*, the average byte accuracy is improved from 72% to 81%. Specifically, Fig.2 shows the byte accuracies achieved for the two most significant applications (*Web*, *Xunlei*). We find that both these two applications’ byte accuracies are improved via *cost-sensitive learning*. One strange phenomenon is that the Peer-to-Peer application *Xunlei* obtained higher byte accuracy rate than the traditional application *Web*, because many current Peer-to-Peer applications disguise themselves by employing the same communicating port and similar payload signature as *Web*. In hand-classification phase, *Web* is labeled via signature-based method, and then some traffic which does not belong to *Web* is labeled as *Web*. This mistake of “base truth” brings interference to the identification of *Web*.

To test the effectiveness of our classification model in real classification process, all traffic that belongs to male/female dormitories is classified with this model. Table 3 shows the classification results. Compared to Table 3, more traffic can be identified. In hand-classification phase, 44% of bytes are regarded as *unknown*, but there are only 23% of bytes that can not be identified with our classification model. It’s worth noting that the number of several application types such as *QQGame* and *MSN* are even less than hand-classification phase, this is because our cost-sensitive strategy

Table 3. Results of Actual Traffic Classification in Male/Female Dormitories

Application Type	Flow		Byte (MB)	
	Amount	Ratio	Amount	Ratio
Web	934011	51.17%	22906.59	20.18%
PPLive	21569	1.18%	330.52	0.29%
Xunlei	231762	12.70%	64230.63	56.57%
FTP	18265	1.00%	4.29	0.00%
MSN	7396	0.41%	8.26	0.01%
QQ	15887	0.87%	53.23	0.05%
QQGame	7471	0.41%	207.26	0.18%
Unknown	589066	32.27%	25791.47	22.72%
Total	1825427	100.00%	113532.25	100.00%

would try to increase the accuracy on the *elephant flows* while potentially decreasing the accuracy on the *mice flows*. Although several mice flows would be ignored, our classification model can obtain a better performance in general.

5 Conclusions

This paper mainly considered two problems when classifying network traffic: the accuracy rate achieved by single classifier is limited, and previous research is mainly focused on achieving only high flow accuracy, while byte accuracy may play a much more important role than flow accuracy in network traffic classification. To address these problems, this paper propose a machine learning framework for classifying network traffic by combining *ensemble learning* with *cost-sensitive learning*.

The aforementioned classification framework is evaluated with a set of real Internet traffic traces. The results show that this method does help to achieve an improvement both in flow accuracy and byte accuracy, with our experiments showing average accuracies of 94% in terms of flows and 81% in terms of bytes can be achieved.

References

1. Sen, S., Spatscheck, O., Wang, D.: Accurate, Scalable In-network Identification of p2p Traffic Using Application Signatures. In: Proceedings of the 13th international conference on World Wide Web, pp. 512–521. ACM, New York (2004)
2. Haffner, P., Sen, S., Spatscheck, O., Wang, D.: ACAS: Automated Construction of Application Signatures. In: Proceedings of the 2005 ACM SIGCOMM workshop on Mining network data, pp. 197–202. ACM, New York (2005)
3. Cache Logic, <http://www.cachelogic.com>
4. Erman, J., Mahanti, A., Arlitt, M., Williamson, C.: Identifying and Discriminating Between Web and Peer to Peer Traffic in the Network Core. In: WWW 2007, Banff, Alberta, Canada (2007)
5. Erman, J., Mahantix, A., Arlitt, M.: Byte Me: A Case for Byte Accuracy in Traffic Classification. In: MineNet 2007, San Diego, California, USA (2007)
6. Dietterich, T.G.: Ensemble Methods in Machine Learning. In: Kittler, J., Roli, F. (eds.) MCS 2000. LNCS, vol. 1857, pp. 1–15. Springer, Heidelberg (2000)
7. Breiman, L.: Bagging Predictors. *Machine Learning* 24, 123–140 (1996)
8. Drummond, C., Holte, R.C.: Exploiting the Cost (In)sensitivity of Decision Tree Splitting Criteria. In: Proceeding of 17th International Conference on Machine Learning, pp. 239–246. Morgan Kaufmann, San Francisco (2000)
9. Fan, W., Stolfo, S.J., Zhang, J., Chan, P.K.: AdaCost: Misclassification Cost-sensitive Boosting. In: Proceedings of 16th International Conference on Machine Learning, pp. 97–105. Morgan Kaufmann, San Francisco (1999)
10. Margineantu, D.D.: Class Probability Estimation and Cost-Sensitive Classification Decisions. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) ECML 2002. LNCS (LNAI), vol. 2430, pp. 270–281. Springer, Heidelberg (2002)
11. Zadrozny, B., Elkan, C.: Learning and Making Decisions When Costs and Probabilities are Both Unknown. In: Proceedings of the seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 204–213. ACM, New York (2001)

12. Domingos, P.: MetaCost: A General Method for Making Classifiers Cost-sensitive. In: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 155–164. ACM, New York (1999)
13. Ribeiro, V.J., Zhang, Z.L., Moon, S., Diot, C.: Small-time Scaling Behavior of Internet Backbone Traffic. *Computer Networks: The International Journal of Computer and Telecommunications Networking* 48, 315–334 (2005)
14. Lan, K.C., Heidemann, J.: A Measurement Study of Correlations of Internet Flow Characteristics. *Computer Networks: The International Journal of Computer and Telecommunications Networking* 50, 46–62 (2006)
15. Quinlan, J.R.: C4.5: Programs for Machine Learning. The Morgan Kaufmann Series in Machine Learning. Morgan Kaufmann, San Mateo (1993)
16. Amit, Y., Geman, D.: Shape Quantization and Recognition with Randomized Trees. *Neural Computation* 9, 1545–1588 (1997)
17. Zhou, Z.H., Wu, J., Tang, W.: Ensembling Neural Networks: Many Could Be Better Than All. *Artificial Intelligence* 137, 239–263 (2002)

Intuitionistic Fuzzy Modules and Their Structures

Chuanyu Xu

Department of Math, Zhejiang Gongshang University
310035 Hangzhou, China
xuchuanyu@yahoo.cn

Abstract. Intuitionistic fuzzy sets ($\mathcal{J}^{\circ}\mathcal{F}\mathcal{S}$) are without any operations. Intuitionistic fuzzy groups ($\mathcal{J}^{\circ}\mathcal{F}\mathcal{G}$) are with a kind of operations based on intuitionistic fuzzy relations. Their algebraic structures are simpler. In order to solve the problem, this paper forwards intuitionistic fuzzy modules ($\mathcal{J}^{\circ}\mathcal{F}\mathcal{M}$ s). They are $\mathcal{J}^{\circ}\mathcal{F}\mathcal{S}$ s with two kinds of operations based on $\mathcal{J}^{\circ}\mathcal{F}$ relations. This paper further studies the external structure of $\mathcal{J}^{\circ}\mathcal{F}\mathcal{M}$ s—the commutative diagram of homomorphism, and two internal structures of $\mathcal{J}^{\circ}\mathcal{F}\mathcal{M}$ s: 1. the free $\mathcal{J}^{\circ}\mathcal{F}\mathcal{M}$ s on the $\mathcal{J}^{\circ}\mathcal{F}\mathcal{R}$ s are the $\mathcal{J}^{\circ}\mathcal{F}$ project modules and the direct sums of $\mathcal{J}^{\circ}\mathcal{F}\mathcal{M}$ s. This paper reveals more algebraic structures of $\mathcal{J}^{\circ}\mathcal{F}\mathcal{S}$ s.

1 Introduction

Intuitionistic fuzzy sets (simply, $\mathcal{J}^{\circ}\mathcal{F}\mathcal{S}$ s) were proposed by Atanassov in 1986 and 1989. [1,2] There exist genuine $\mathcal{J}^{\circ}\mathcal{F}\mathcal{S}$ s which are not fuzzy sets. [1] For $\mathcal{J}^{\circ}\mathcal{F}\mathcal{S}$, the logical law of the excluded middle is not valid.[1] However $\mathcal{J}^{\circ}\mathcal{F}\mathcal{S}$ s are without any operation. Recently, intuitionistic fuzzy groups (simply $\mathcal{J}^{\circ}\mathcal{F}\mathcal{G}$ s) with operations based on $\mathcal{J}^{\circ}\mathcal{F}$ relations were proposed. [3-5] The algebraic systems are the elements with operations. So the algebraic systems of $\mathcal{J}^{\circ}\mathcal{F}\mathcal{R}$ s are more complex than $\mathcal{J}^{\circ}\mathcal{F}\mathcal{G}$ s. However there is only a kind of operation for any $\mathcal{J}^{\circ}\mathcal{F}\mathcal{R}$. Therefore its algebraic structure is still simpler. It is necessary to study the $\mathcal{J}^{\circ}\mathcal{F}\mathcal{S}$ with two kinds of operations based on $\mathcal{J}^{\circ}\mathcal{F}$ relations.

This paper forwards intuitionistic fuzzy rings ($\mathcal{J}^{\circ}\mathcal{F}\mathcal{R}$ s) and intuitionistic fuzzy modules ($\mathcal{J}^{\circ}\mathcal{F}\mathcal{M}$ s). They are $\mathcal{J}^{\circ}\mathcal{F}\mathcal{S}$ s with two kinds of operations based on $\mathcal{J}^{\circ}\mathcal{F}$ relations. This paper further studies the external structure of $\mathcal{J}^{\circ}\mathcal{F}\mathcal{M}$ s—the commutative diagram of homomorphism (Theorem 1), and two internal structures of $\mathcal{J}^{\circ}\mathcal{F}\mathcal{M}$ s: 1. the free $\mathcal{J}^{\circ}\mathcal{F}\mathcal{M}$ s on the $\mathcal{J}^{\circ}\mathcal{F}\mathcal{R}$ s are the $\mathcal{J}^{\circ}\mathcal{F}$ project modules (Theorem 2), and 2. the direct sums of $\mathcal{J}^{\circ}\mathcal{F}\mathcal{M}$ s (Theorem 3).

The difference between the related work and this paper is as follows. In the available $\mathcal{J}^{\circ}\mathcal{F}\mathcal{R}$ s and $\mathcal{J}^{\circ}\mathcal{F}\mathcal{M}$ s, their sets and operations are ones in classical groups. However, in this paper, the sets of $\mathcal{J}^{\circ}\mathcal{F}\mathcal{R}$ s and the $\mathcal{J}^{\circ}\mathcal{F}\mathcal{M}$ s are $\mathcal{J}^{\circ}\mathcal{F}\mathcal{S}$ s, and their two kinds

of operations are based on $\mathcal{J}^{\circ}F$ relations. [5] This paper reveals more algebraic structures of $\mathcal{J}^{\circ}FS$ s.

The rest of paper is organized as follows: 2.preliminaries, 3. Intuitionistic Fuzzy Rings and Modules 4. Structures of $\mathcal{J}^{\circ}FM$ s, and 5.Conclusion.

2 Preliminaries

Definition 2.1 [1,2] (Intuitionistic Fuzzy Set) Let a set E be fixed. An IFS A in E is an object having the form

$$A = \{ \langle x, \mu_A(x), \nu_A(x) \rangle \mid x \in E \},$$

where the functions $\mu_A(x):E \rightarrow [0,1]$ and $\nu_A(x):E \rightarrow [0,1]$ define the degree of membership and the degree of nonmembership of the element $x \in E$ to the set A , which is a subset of E , respectively, and for every $x \in E$:

$$0 \leq \mu_A(x) + \nu_A(x) \leq 1.$$

Note. Obviously every fuzzy set has the form

$$\{ \langle x, \mu_A(x), 1 - \mu_A(x) \rangle \mid x \in E \}.$$

Definition 2.2 [4,5] (Intuitionistic Fuzzy binary operation). Let $\theta \in [0,1]$, G be a non-void set, and μ and ν be two non-negative mappings from $G \times G \times G$ to $[0,1]$, respectively, if

$$(1) \forall a, b \in G, \exists c \in G, \text{ such that } \mu(a, b, c) > \theta \text{ and } \nu(a, b, c) > \theta;$$

(2) $\forall a, b \in G, \forall c_1, c_2 \in Y, \mu(x, c_1) > \theta$ and $\nu(x, c_1) > \theta; \mu(x, c_2) > \theta$ and $\nu(x, c_2) < 1 - \theta \Rightarrow c_1 = c_2$, then the vector function (μ, ν) is called an intuitionistic Fuzzy binary operation, mapping denoted as $\langle \mu, \nu \rangle$.

Definition 2.3 [9,10] (Fuzzy binary operation). Let $\theta \in (0,1)$. If mapping $R: G \times G \times G \rightarrow [0,1]$ satisfies:

$$(1) \forall a, b \in G, \exists c \in G, R(a, b, c) > \theta;$$

$$(2) R(a, b, c_1) \wedge R(a, b, c_2) > \theta \Rightarrow c_1 = c_2,$$

then R is called a Fuzzy binary operation on G .

From the definition 2.1 and its note, contrast the definition 2.2 with the definition 2.3, it is saw that if the definition 2.2 takes the form of $\mathcal{J}^{\circ}FS$, the definition 2.2 is the same as the definition 2.2. Therefore, it is in accordance with the definition of genuine Intuitionistic Fuzzy binary operation that if $\mu(a, b, c) > \theta_1$ and $\nu(a, b, c) > \theta_2$, here $\theta_1, \theta_2 \in [0,1]$, were taken.

Correspondent relation defines mapping. Firstly, Intuitionistic Fuzzy relation is shown.

Definition 2.4(Intuitionistic Fuzzy relation, $\mathcal{J}^{\circ}FR$) Let X and Y be two nonvoid sets, an Intuitionistic Fuzzy set in $X \times Y$ is called an Intuitionistic Fuzzy relation from X to Y .

Note. $\mathcal{J}^\circ\text{FR}$ is the form:

$$\{ \langle (x, y), \mu(x, y), \nu(x, y) \rangle \mid (x, y) \in X \times Y \},$$

where $\mu(x, y) > \theta_1$ and $\nu(x, y) > \theta_2$ satisfy the conditions of definition 2.1.

$\mathcal{J}^\circ\text{F}$ mapping is defined based on $\mathcal{J}^\circ\text{FR}$.

Definition 2.5 (Intuitionistic Fuzzy mapping). Let X and Y be two nonvoid sets, there is an $\mathcal{J}^\circ\text{FR}$ on $X \times Y$, and $\exists \theta_1, \theta_2 \in [0, 1]$. If

- (1) $\forall x \in X, \exists y \in Y$, such that $\mu(x, y) > \theta_1$ and $\nu(x, y) > \theta_2$
- (2) $\forall x \in X, \forall y_1, y_2 \in Y, \mu(x, y_1) > \theta_1$ and $\nu(x, y_1) > \theta_2, \mu(x, y_2) > \theta_1$ and $\nu(x, y_2) > \theta_2 \Rightarrow y_1 = y_2$,

then the vector function (μ, ν) is called an intuitionistic Fuzzy mapping ($\mathcal{J}^\circ\text{FM}$) $(\mu, \nu): X \rightarrow Y, x \mapsto y$, denoted as $(\mu, \nu)(x) = y$, or for simplicity, $\varphi(x) = y$. |

In order to compare two $\mathcal{J}^\circ\text{F}$ Ss with each other, it is necessary to further define $\mathcal{J}^\circ\text{F}$ mapping.

Definition 2.6. If $\mathcal{J}^\circ\text{FM} (\mu, \nu)$ satisfies that $\forall y \in Y, \exists x \in X$ such that $\mu(x, y) > \theta_1, \nu(x, y) > \theta_2$, then (μ, ν) is called $\mathcal{J}^\circ\text{F}$ surjection. If $\forall x_1, x_2 \in X, \forall y \in Y, \mu(x_1, y) > \theta_1, \nu(x_1, y) > \theta_2$, and $\mu(x_2, y) > \theta_1, \nu(x_2, y) > \theta_2 \Rightarrow x_1 = x_2$, then (μ, ν) is called $\mathcal{J}^\circ\text{F}$ injection. If (μ, ν) is both $\mathcal{J}^\circ\text{F}$ surjection and $\mathcal{J}^\circ\text{F}$ injection, then (μ, ν) is called $\mathcal{J}^\circ\text{F}$ bijection. |

In the following definition, if $\forall x, y \in G, (x, y) \in G \times G$, and $\exists z \in G$ such that $(\mu, \nu)(x, y) = z$ then it is a $\mathcal{J}^\circ\text{F}$ relation from $(x, y) \in G \times G$ to $z \in G$. $\mathcal{J}^\circ\text{F}$ Binary operation is shown. below

Definition 2.7 [4,5 9-11]($\mathcal{J}^\circ\text{F}$ Binary operation). Let G be a nonvoid set, and $\mathcal{J}^\circ\text{F}$ M, $(\mu, \nu): G \times G \times G \rightarrow [0, 1]$, respectively. If

- (1) $\forall x, y \in G, \exists z \in G$, such that $\mu(x, y, z) > \theta_1$ and $\nu(x, y, z) > \theta_2$;
- (2) $\forall x, y \in G, \forall z_1, z_2 \in G, \mu(x, y, z_1) > \theta_1, \nu(x, y, z_1) > \theta_2$; and $\mu(x, y, z_2) > \theta_1, \nu(x, y, z_2) > \theta_2 \Rightarrow z_1 = z_2$, then the vector function (μ, ν) is called an $\mathcal{J}^\circ\text{F}$ binary operation on G .

Denote $(x \circ y)(z) \triangleq \langle \mu(x, y, z), \nu(x, y, z) \rangle$, here ‘ \circ ’ is called the $\mathcal{J}^\circ\text{F}$ binary operator. |

If the operation between two elements is indirect, and can be established by another element, it is the composition operation as follows.

Definition 2.8. The $\mathcal{J}^\circ\text{F}$ composition operation between elements in G is defined as follows:

$$\begin{aligned} & ((x \circ y) \circ z)(a) \\ &= \langle \bigvee_{b \in G} (\mu(x, y, b) \wedge \mu(b, z, a)), \bigwedge_{c \in G} (\nu(x, y, c) \vee \nu(c, z, a)) \rangle \triangleq \langle \mu^*, \nu^* \rangle \\ & \quad (x \circ (y \circ z))(a) \\ &= \langle \bigvee_{b \in G} (\mu(y, z, b) \wedge \mu(x, b, a)), \bigwedge_{c \in G} (\nu(x, y, c) \vee \nu(x, c, a)) \rangle \triangleq \langle \mu^*, \nu^* \rangle \end{aligned} |$$

The next definition shows $\mathcal{F}^{\circ}F$ group.

Definition 2.9 [1,2,4,5]($\mathcal{F}^{\circ}F$ group) Let G be a nonvoid set. If

- (1) $((x \circ y) \circ z)(a_1) = \langle \mu^*, \nu^* \rangle, (x \circ (y \circ z))(a_2) = \langle \mu^*, \nu^* \rangle, \mu^*, \nu^* > \theta_1, \nu^*, \nu^* > \theta_2 \Rightarrow a_1 = a_2$, ' \circ ' is called to satisfy the association law;
- (2) $\forall x \in G, \exists e \in G, (e \circ x)(x) = \langle \mu(e, x, x), \nu(e, x, x) \rangle, (x \circ e)(x) = \langle \mu(x, e, x), \nu(x, e, x) \rangle, \mu(\bullet, \bullet, \bullet) > \theta_1, \nu(\bullet, \bullet, \bullet) > \theta_2$, e is called an identity element;
- (3) $\forall x \in G, \exists y \in G, (x \circ y)(e) = \langle \mu(x, y, e), \nu(x, y, e) \rangle, (y \circ x)(e) = \langle \mu(y, x, e), \nu(y, x, e) \rangle, \mu(\bullet, \bullet, \bullet) > \theta_1, \nu(\bullet, \bullet, \bullet) > \theta_2$, y is called an inverse element of x , and denoted as x^{-1} ;

then G is called a $\mathcal{F}^{\circ}F$ group. |

In order to study $\mathcal{F}^{\circ}FG$ s by means of subset, $\mathcal{F}^{\circ}F$ subgroup is defined in the next definition.

Definition 2.10[1,2,4,5] ($\mathcal{F}^{\circ}F$ subgroup) If the nonempty subset H of $\mathcal{F}^{\circ}F$ group G is a $\mathcal{F}^{\circ}F$ group about the operation ' \circ ', then H is called a $\mathcal{F}^{\circ}F$ subgroup of G , denoted by $H < G$. |

In order to sort $\mathcal{F}^{\circ}FG$ s by means of $\mathcal{F}^{\circ}F$ subgroup, it is necessary to define coset and $\mathcal{F}^{\circ}F$ equivalent relation as follows.

Definition 2.11. [1,2,4,5] Suppose that H is an $\mathcal{F}^{\circ}F$ subgroup of $\mathcal{F}^{\circ}FG$, $x, z \in G$, define

$$\begin{aligned} (xH)(z) &= \langle \bigvee_{h \in H} \mu(x, h, z), \bigwedge_{h \in H} \nu(x, h, z) \rangle \\ (Hx)(z) &= \langle \bigvee_{h \in H} \mu(h, x, z), \bigwedge_{h \in H} \nu(h, x, z) \rangle \end{aligned}$$

xH and Hx are called the $\mathcal{F}^{\circ}F$ left coset and the $\mathcal{F}^{\circ}F$ right coset of H in x , respectively. |

3 Intuitionistic Fuzzy Rings and Modules

Definition 3.1. Let R be a nonvoid set. If

- (1) R is additive Abel group about a binary operation called addition;
- (2) the another binary operation called the multiplication on R is defined, the multiplication satisfies the combination

$$((a \circ b) \circ c)(z) = (a \circ (b \circ c))(z),$$

and the left and right distributive laws:

$$(a \circ (b + c))(z) = (a \circ b + a \circ c)(z),$$

where $a, b, c, z \in R$, then R is called Intuitionistic Fuzzy Rings ($\mathcal{F}^{\circ}FR$ s).

If the multiplication of $\mathcal{F}^{\circ}F$ -ring satisfies commutative law, i.e.,

$$a \circ b = b \circ a, \quad a, b \in R$$

then R is called the $\mathcal{F}^{\circ}F$ commutative ring.

If there is an identity in the multiplication of R , i.e., then $\forall a \in R, 1 \circ a = a \circ 1 = a$, then R is called $\mathcal{J}^{\circ}F R$ with an identity. |

Definition 3.2. Let R be an $\mathcal{J}^{\circ}F R$, and M be an additive Abel $\mathcal{J}^{\circ}F G$. If there exists $\mathcal{J}^{\circ}F$ -mapping $R \times M \rightarrow M, (a, x) \mapsto ax$, where ax is the image of (a, x) , which satisfies

- (1) $a(x+y) = ax + ay, a \in R, x, y \in M$;
- (2) $(a+b)x = ax + bx, a, b \in R, x \in M$;
- (3) $(a \circ b)x = a \circ (bx), a, b \in R, x \in M$;
- (4) $1x = x, x \in M, 1 \in R$ is the identity of multiplication, then M is called an Intuitionistic Fuzzy Module ($\mathcal{J}^{\circ}F M$) on $\mathcal{J}^{\circ}F R$. |

Definition 3.3. Let N be the nonvoid set and R be the $\mathcal{J}^{\circ}F R$. If

- (1) N is the $\mathcal{J}^{\circ}F$ additive subgroup;
- (2) for any $a \in R, x \in N, ax \in N$.

then N is called the $\mathcal{J}^{\circ}F$ submodule. |

$\{0\}$ and M themselves are the $\mathcal{J}^{\circ}F$ -submodules of M , they are the trivial $\mathcal{J}^{\circ}F$ -submodules. Except them, the $\mathcal{J}^{\circ}F M$ without any true $\mathcal{J}^{\circ}F$ -submodule is called the $\mathcal{J}^{\circ}F$ -simple submodule.

Definition 3.4. Let $x_1, \dots, x_n \in M$ and R is the $\mathcal{J}^{\circ}F$ -ring, then, $N = \{\sum_{i=1}^n a_i x_i | a_i \in R, 1 \leq i \leq n\}$ is the $\mathcal{J}^{\circ}F$ -finite generating submodule of M . If $M = N$, then M is the $\mathcal{J}^{\circ}F$ -finite generating module and $\{x_1, \dots, x_n\}$ are the generators of M . |

Definition 3.5. Let R is the $\mathcal{J}^{\circ}F$ -ring. If the $\mathcal{J}^{\circ}F$ -finite generators are free which mean

$$r_1 x_1 + \dots + r_s x_s = 0 \in M, r_1, \dots, r_s \in R \text{ iff } r_1 = \dots = r_s = 0,$$

where 0 is the identity of $\mathcal{J}^{\circ}F$ -module, then M is called $\mathcal{J}^{\circ}F M$, and the generators are called the base. |

Definition 3.6. For the two $\mathcal{J}^{\circ}F M$ s M and M' , if there exists $\mathcal{J}^{\circ}F$ -mapping $\varphi: M \rightarrow M'$ such that

- (1) $\varphi(x+y) = \varphi(x) + \varphi(y), x, y \in M$;
- (2) $\varphi(ax) = a\varphi(x), a \in R, x \in M$,

then φ is called the homomorphism of $\mathcal{J}^{\circ}F M$ s from M to M' . The set of all the homomorphisms of $\mathcal{J}^{\circ}F M$ s denoted as $\text{Hom}_R(M, M')$. |

Definition 3.7. Let φ be the homomorphism of $\mathcal{J}^{\circ}F$ -modules. Taking for the $\mathcal{J}^{\circ}F$ -mapping of sets, if φ is the injection, the surjection and the bijection, then φ is called the monomorphism, the epimorphism and the isomorphism of $\mathcal{J}^{\circ}F$ -modules, respectively. |

Definition 3.8. If S and A are sets and $S \subset A$, then $\mathcal{G}^{\circ F}$ -mapping $\mathbf{1}_{A|S}: S \rightarrow A$ is called the inclusion mapping from S to A . |

4 Structures of $\mathcal{G}^{\circ F}$ Ms

The relation between the homomorphism of $\mathcal{G}^{\circ F}$ -modules is the external structure of $\mathcal{G}^{\circ F}$ Ms. For the homomorphism from the $\mathcal{G}^{\circ F}$ Ms with base to any $\mathcal{G}^{\circ F}$ Ms, it is certain to find such the third $\mathcal{G}^{\circ F}$ M as to build the commutative diagram of homomorphism. This is the theorem 1.

Theorem 1. Let R be the $\mathcal{G}^{\circ F}R$ and M be the $\mathcal{G}^{\circ F}M$. The proposition (1) derives the proposition (2):

(1) there is $\mathcal{G}^{\circ F}$ -base of M ;

(2) there exists a nonvoid set X and the $\mathcal{G}^{\circ F}$ -mapping $\xi: X \rightarrow M$, which possesses the property that for any $\mathcal{G}^{\circ F}M$ and $\mathcal{G}^{\circ F}$ -mapping $f: X \rightarrow A$, there is the unique homomorphism of $\mathcal{G}^{\circ F}$ Ms $\bar{f}: M \rightarrow A$, such that $\bar{f}\xi=f$.

Proof

Let X be the base of $\mathcal{G}^{\circ F}M$. Then $\xi: X \rightarrow M$ is the the inclusion mapping. If $u \in M$, then X spans M If.

$$u = \sum_{i=1}^n r_i x_i (r_i \in R, x_i \in X)$$

And

$$u =_{\text{def}} \sum_{i=1}^n s_i x_i (s_i \in R, x_i \in X),$$

then

$$\sum_{i=1}^n (x_i - s_i) x_i = 0,$$

From the independence of the base. $\forall i, x_i = s_i$ the $\mathcal{G}^{\circ F}$ -mapping can be defined as

$$\bar{f}: M \rightarrow A, \quad \bar{f}(u) = \bar{f}\left(\sum_{i=1}^n r_i x_i\right) = \sum_{i=1}^n r_i \bar{f}(x_i)$$

And $\bar{f}\xi = f$. It is necessary to test and verify that \bar{f} is the homomorphism of $\mathcal{G}^{\circ F}$ -modules. In fact, X spans M , therefore every homomorphism of $\mathcal{G}^{\circ F}$ -modules, $M \rightarrow A$, is uniquely determined by its action on X . If $g: M \rightarrow A$ is the homomorphism of $\mathcal{G}^{\circ F}$ -modules and $g\xi=f$, then $\forall x \in X$,

$$g(x) = g(\xi(x)) = f(x) = \bar{f}(x),$$

so as to $g = \bar{f}$, that is, \bar{f} is unique. |

The $\mathcal{G}^{\circ F}$ project module is important in indicating the internal structure of $\mathcal{G}^{\circ F}$ Ms and the commutative diagram of homomorphism of $\mathcal{G}^{\circ F}$ Ms.

Definition 4.1 ($\mathcal{J}^{\circ F}$ project modules). Let P be the homomorphism of $\mathcal{J}^{\circ F}$ Ms. If \forall the epimorphism of $\mathcal{J}^{\circ F}$ Ms $\varepsilon: B \rightarrow C$ and the homomorphism of $\mathcal{J}^{\circ F}$ Ms $\gamma: P \rightarrow C$, there exists the homomorphism of $\mathcal{J}^{\circ F}$ Ms $\beta: P \rightarrow B$ such that $\varepsilon \bullet \beta = \gamma$, then P is called the $\mathcal{J}^{\circ F}$ project module. |

The relation between the free $\mathcal{J}^{\circ F}$ Ms and the $\mathcal{J}^{\circ F}$ -project modules is as follows.

Theorem 2. The free $\mathcal{J}^{\circ F}$ M on the $\mathcal{J}^{\circ F}$ R is the $\mathcal{J}^{\circ F}$ project module.

Proof

Suppose that there is the homomorphism diagram of $\mathcal{J}^{\circ F}$ Ms, where ε is the epimorphism of $\mathcal{J}^{\circ F}$ Ms and M is the free $\mathcal{J}^{\circ F}$ M on the set X . Therefore, $\iota: X \rightarrow M$. $\forall x \in X, \gamma(\iota(x)) \in C$. $\therefore \varepsilon$ is the epimorphism of $\mathcal{J}^{\circ F}$ -modules, \therefore for $\gamma(\iota(x)) \in C$, there must exist $a_x \in A$ such that $\varepsilon(a_x) = \gamma(\iota(x))$. Because M is the free $\mathcal{J}^{\circ F}$ M, $\mathcal{J}^{\circ F}$ -mapping $X \rightarrow A$, $x \mapsto a_x$ induces the homomorphism diagram of $\mathcal{J}^{\circ F}$ Ms $\beta: M \rightarrow A$ such that $\forall x \in X$,

$$\beta(\iota(x)) = a_x, \varepsilon \bullet \beta \bullet \iota(x) = \varepsilon(a_x) = \gamma \bullet \iota(x),$$

therefore $\varepsilon \bullet \beta \bullet \iota = \gamma \bullet \iota: X \rightarrow B$ is the epimorphism. $\therefore M$ is the free $\mathcal{J}^{\circ F}$ M, therefore $\iota: X \rightarrow M$. $\forall x \in X, \gamma(\iota(x)) \in C$. Because ε is the $\mathcal{J}^{\circ F}$ -epimorphism, for $\gamma(\iota(x)) \in C$, there must be $a_x \in A$ such that $\varepsilon(a_x) = \gamma(\iota(x))$. $\therefore M$ is $\mathcal{J}^{\circ F}$ -free module, $\therefore \mathcal{J}^{\circ F}$ -mapping $X \rightarrow A$, $x \mapsto a_x$ induces the homomorphism of $\mathcal{J}^{\circ F}$ -modules $\beta: M \rightarrow A$ such $\forall x \in X$, $\beta(\iota(x)) = a_x$, as that

$$\varepsilon \bullet \beta \bullet \iota(x) = \varepsilon(a_x) = \gamma \bullet \iota(x),$$

therefore

$$\varepsilon \bullet \beta \bullet \iota = \gamma \bullet \iota: X \rightarrow B.$$

From the uniqueness of the theorem1, it is obtained that $\varepsilon \bullet \beta = \gamma$. Therefore M is the project module. |

Definition 4.2. Let N be the $\mathcal{J}^{\circ F}$ -submodule of $\mathcal{J}^{\circ F}$ M M , specify the action of $\mathcal{J}^{\circ F}$ R on $\mathcal{J}^{\circ F}$ -quotient group M/N as follows:

$$r(x+N) = rx + M, \forall r \in R, x \in M,$$

then M/N is called the quotient module of M about the $\mathcal{J}^{\circ F}$ -submodule N . |

The direct sum of $\mathcal{J}^{\circ F}$ M is the important structural property. They are discussed below.

Definition 4.3. Let N and P be the $\mathcal{J}^{\circ F}$ -submodule of the $\mathcal{J}^{\circ F}$ M M such that

- (1) $M = N + P$;
- (2) $N \cap P = \{0\}$, 0 is the identity of the addition of $\mathcal{J}^{\circ F}$ Ms,

then M is called the direct sum of N and P , denoted as $M = N \oplus P$. |

Which $\mathcal{J}^{\circ}F M$ does include the $\mathcal{J}^{\circ}F$ -project module? Are equivalent the $\mathcal{J}^{\circ}F$ -project module and the commutative diagram? The answer is seen below.

Theorem 3. For $\mathcal{J}^{\circ}F$ -finite generating module P , the following propositions are equivalent:

- (1) P is the $\mathcal{J}^{\circ}F$ -project module;
- (2) In the homomorphism diagram of $\mathcal{J}^{\circ}F M$, for $\varepsilon : B \rightarrow C$ there is the epimorphism of Abel $\mathcal{J}^{\circ}F M$ s

$$\varepsilon_* : \text{Hom}_R(P, B) \rightarrow \text{Hom}_R(P, C),$$

where $\varepsilon_*(\beta) = \varepsilon \bullet \beta, \beta \in \text{Hom}_R(P, B)$;

- (3) If $\varepsilon : B \rightarrow P$ is the epimorphism of $\mathcal{J}^{\circ}F M$ s, then there exists the homomorphism diagram of $\mathcal{J}^{\circ}F M$ s such $\beta : P \rightarrow B$ that $\varepsilon \bullet \beta = 1_P$;
- (4) If $P \cong B/A$ where $A = \ker \varepsilon$, then P is the term of the direct sum;
- (5) P is the term of the direct sum of some $\mathcal{J}^{\circ}F$ -finite generating module.

Proof

(1) \Rightarrow (2)

From the definition of $\mathcal{J}^{\circ}F$ -project module, and the epimorphism of $\mathcal{J}^{\circ}F M$ s $\varepsilon : B \rightarrow C$, for any the homomorphism of $\mathcal{J}^{\circ}F M$ s $\gamma : P \rightarrow C$, there must exist the homomorphism of $\mathcal{J}^{\circ}F M$ s $\beta : P \rightarrow B$, therefore it is determined that $\varepsilon_* : \text{Hom}_R(P, B) \rightarrow \text{Hom}_R(P, C)$.

(2) \Rightarrow (3)

Known that the epimorphism of $\mathcal{J}^{\circ}F M$ s, $\varepsilon : B \rightarrow C$. From (2), ε_* is the epimorphism of $\mathcal{J}^{\circ}F M$ s. Take $C = P$. Then for the homomorphism $(P \rightarrow P) \in \text{Hom}_R(P, P)$, there exists the homomorphism $(P \rightarrow B) \in \text{Hom}_R(P, B)$ such that $\text{Hom}_R(P, B) \rightarrow \text{Hom}_R(P, P)$. Specially, $1_P \in \text{Hom}_R(P, P)$. For 1_P , there is $(P \rightarrow B)$ such that $(P \rightarrow B) \mapsto 1_P$.

(3) \Rightarrow (4)

(a) Let $\varepsilon : B \rightarrow P$ be the epimorphism of $\mathcal{J}^{\circ}F M$ s. From (3), there must be the homomorphism of $\mathcal{J}^{\circ}F M$ s $\beta : P \rightarrow B$ such that $\varepsilon \bullet \beta = 1_P$. $P' \triangleq \text{Im} \beta \subset B$. $\varepsilon \bullet \beta = 1_P \Rightarrow \varepsilon(P') = P$. $\ker \varepsilon = \{0\} \subset P$. Now, for $\beta : P \rightarrow P'$, and from (3), $\varepsilon \bullet \beta = 1_P \Rightarrow P \rightarrow P' \rightarrow P$, $\therefore \beta \bullet \varepsilon = 1_{P'}$, $\Leftrightarrow P' \cong P$. $\therefore \varepsilon : P' \rightarrow P$ is the isomorphism of $\mathcal{J}^{\circ}F M$ s, $\therefore \ker \varepsilon = \{0\} \subset P'$. On the other hand, from the homomorphism theorem of $\mathcal{J}^{\circ}F G[5]$, $\ker \varepsilon = A$, $\therefore P' \cap A = \{0\}$.

(b) Let $b \in B$, then $b \in B/A$. Known that under $\varepsilon, P \cong B/A$, $\therefore \varepsilon(b) \in A$. And known that $\varepsilon : P' \rightarrow P$ is the isomorphism of $\mathcal{J}^{\circ}F M$ s., $\therefore \exists x \in P'$ such that $\varepsilon(b) = \varepsilon(x)$, thus $a = b - x \in \ker \varepsilon$, that is $b = a + x$, Therefore $B = P' \oplus A$, P is the term of the direct sum of B .

(4) \Rightarrow (5)

$\therefore P \cong M/A$, $\therefore P$ is the term of the direct sum of M .

(5) \Rightarrow (1)

Let $V = P \oplus K$. Considering that the epimorphism of $\mathcal{J}^{\circ}F M$ s $\pi : V \rightarrow P$, $(p, k) \mapsto p$ and the monomorphism of $\mathcal{J}^{\circ}F M$ s $\iota : P \rightarrow V$, $p \mapsto (p, 0)$, where $p \in P$, $k \in K$. Obviously,

$\pi \bullet \iota = \mathbf{1}_P$. From the theorem 2, the free $\mathcal{J}^{\circ F}M$ is the $\mathcal{J}^{\circ F}$ project module., there is the homomorphism of $\mathcal{J}^{\circ F}M \beta': M \rightarrow B$ such that

$$\varepsilon \bullet \beta' = \gamma \bullet \pi. \text{ Let } \beta = \beta' \bullet \iota: P \rightarrow B,$$

then

$$\varepsilon \bullet \beta = \varepsilon \bullet (\beta' \bullet \iota) = (\varepsilon \bullet \beta') \bullet \iota = (\gamma \bullet \pi) \bullet \iota = \gamma \bullet (\pi \bullet \iota) = \gamma \mathbf{1}_P = \gamma. \quad |$$

5 Conclusions

This paper forwards Intuitionistic Fuzzy Rings ($\mathcal{J}^{\circ F}Rs$) and Intuitionistic Fuzzy Modules ($\mathcal{J}^{\circ F}Ms$). They are $\mathcal{J}^{\circ F}Ss$ with two kinds of operations based on $\mathcal{J}^{\circ F}$ relations. This paper further studies the external structure of $\mathcal{J}^{\circ F}Ms$ —the commutative diagram of homomorphism (Theorem 1), and two internal structures of $\mathcal{J}^{\circ F}Ms$: 1.the free $\mathcal{J}^{\circ F}Ms$ on the $\mathcal{J}^{\circ F}Rs$ are the $\mathcal{J}^{\circ F}$ project modules (Theorem 2), and 2. the direct sums of $\mathcal{J}^{\circ F}Ms$ (Theorem 3). This paper enriches the structure of $\mathcal{J}^{\circ F}Ss$.

References

1. Atanassov, K.T.: Intuitionistic Fuzzy Sets. Fuzzy sets and systems 20, 87–96 (1986)
2. Atanassov, K.T.: More on Intuitionistic Fuzzy Sets. Fuzzy sets and systems 33, 37–45 (1989)
3. Li, X.P., Wang, G.J.: Intuitionistic Fuzzy Group and its Homomorphic Image. Fuzzy systems and mathematics 14(1), 45–50 (2000)
4. Ban, X.G., Liu, F.: Intuitionistic Fuzzy Group Based on Bi-factor operation. Fuzzy systems and mathematics 20(4), 16–21 (2006)
5. Ban, X.G., Liu, F.: The Sub-Intuitionistic Fuzzy Group and Normal Sub-Intuitionistic of Intuitionistic Fuzzy Group. Fuzzy systems and mathematics 20(3), 43–46 (2006)
6. Cao, X.H., Ye, J.C.: Representation Theory of Groups. Peking University Press, Beijing (1998)
7. Cao, X.H.: Basis of Finite Groups Theory. Higher Education Press, Beijing (1992)
8. Wang, E.F.: Basis of Finite Groups. Qinghua University Press, Beijing (2002)
9. Yuan, X.H., Ren, Y.H., Lin, L.: A New Kind of Fuzzy Group. Journal of Liaoning Normal University (Natural Science Edition) 25(1), 3–6 (2002)
10. Yuan, X.H., Zhang, Y.H., Yang, J.H.: Homomorphism of Fuzzy Group. Journal of Liaoning Normal University (Natural Science Edition) 25(4), 340–342 (2002)

Optimization of Ni-MH Battery Fast Charging in Electric Vehicles Using Dynamic Data Mining and ANFIS

Guifang Guo, Peng Xu, Zhifeng Bai, Shiqiong Zhou,
Gang Xu, and Binggang Cao

School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an 710049 China

Abstract. Fast and efficient charging of Ni-MH battery is a problem which is difficult and often expensive to solve using conventional techniques. This study proposes a method that the integrated data mining algorithm and the Adaptive Network Fuzzy Inference Systems (ANFIS) for discovering the fast charging more efficiently and presenting it more concisely. Because the battery charging is a highly dynamic process, dynamic data mining technique is used for extracting of control rules for effective and fast battery process. The ideal fast charging current has been obtained. The result indicates that the integrated method of adaptive charging current has effectively improved charging efficiency and avoided overcharge and overheating.

Keywords: fast charging; data mining; adaptive network fuzzy inference systems (ANFIS).

1 Introduction

To reduce environmental pollution and to reduce vehicles' fuel consumption, electric vehicles (EVs) may be the optimal way. Ni-MH battery is widely favorable used in EVs, largely because of better recharge ability with almost no memory effect, good cycle life and its high specific energy. Therefore, fast charging of Ni-MH battery is a key technology for EVs industry. That is, fast charge remains a real challenge for the Ni-MH traction battery in EVs. Unlike conventional constant current or constant voltage, the fast intelligent charger should monitor the battery parameters continuously, and alter charge current as frequently as required to prevent overcharging, to prevent exceeding temperature limits, and so on.

Fast charging presents certain merits for all types of the traction batteries. One obvious reason is: the shorter the recharge time, the more convenience the EVs operation. Therefore, the development of effective charge control algorithms not only can reduce charging time, increase energy efficiency but also can improve the market acceptance of EVs. Therefore, searching for an appropriate fast charging control model is essential for Ni-MH battery.

As well known, the battery charging process is a highly dynamic nonlinear process, so the fast charging process couldn't efficiently be managed. The problem of fast charging has been studied extensively in the literature [1-4]. But little progress has been made

in creating of intelligent control system of battery fast charging that provides the features of adaptability and the flexible nature and learning from experiments [5].

To design the required fast intelligent charger, there is a need for extraction of efficient rule base. The rule base should be flexible and should be adjusted on-line during the life of the battery. Data mining technique is known as “knowledge discovery in databases” [6] which can extract the charging rules from the large databases, especially when the charging logic is to be continuously changed during the life of the battery. Moreover, the highly dynamic chemical processes in battery would require application of ANFIS proven to be more efficient.

Therefore, in this study a integrated dynamic data mining and ANFIS model has been presented. ANFIS trained by genetic algorithms which generate a fuzzy rule base to control battery fast charging process.

2 ANFIS and Data Mining

2.1 ANFIS Theory

The general structure of ANFIS with two inputs x and y and one output z is shown in Fig.1. Assume a rule base with two fuzzy if-then rules as follows:

Rule 1: If x is A_1 and y is B_1 then $z_1 = p_1x + q_1y + r_1$;

Rule 2: If x is A_2 and y is B_2 then $z_2 = p_2x + q_2y + r_2$.

where A_1, A_2 and B_1, B_2 are fuzzy sets of input premise variables x and y respectively; p_1, q_1, r_1 and p_2, q_2, r_2 are parameters of the consequent. The final output of network is as follows:

$$\begin{aligned}
 z &= \frac{w_1}{w_1 + w_2} f_1 + \frac{w_2}{w_1 + w_2} f_2 = \bar{w}_1 f_1 + \bar{w}_2 f_2 \tag{1} \\
 &= \bar{w}_1 (p_1x + q_1y + r_1) + \bar{w}_2 (p_2x + q_2y + r_2) \\
 &= (\bar{w}_1x) p_1 + (\bar{w}_1y) q_1 + (\bar{w}_1) r_1 + (\bar{w}_2x) p_2 + (\bar{w}_2y) q_2 + (\bar{w}_2) r_2
 \end{aligned}$$

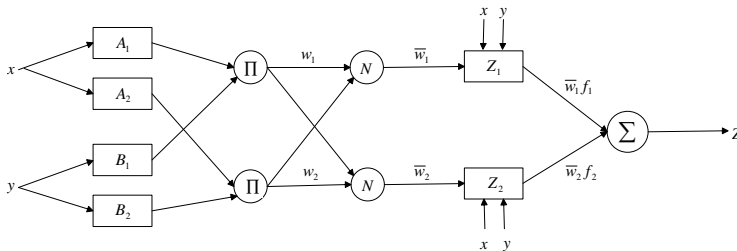


Fig. 1. ANFIS structure

2.2 Data Mining

Data mining is the process of discovering meaningful patterns in huge databases. It is also an application that can provide significant competitive advantages for making the right decision [7-8]. The rule extraction is one of the major forms of data mining.

As it is crucial in ANFIS to have the number of fuzzy rules as minimum as possible, a subtractive clustering method [9] is used whose parameters would control the number of fuzzy rules. In subtractive clustering, the potential of each data point to be a cluster center is estimated and the points with high potential values are selected as candidates for cluster centers. Taking k data points x_k , $k=1, 2, \dots, n$, each of which is a d dimensional vector, the potential of x_k to be a cluster center may be estimated as below:

$$P_k = \sum_{j=1}^n \exp \left(-4 \sqrt{\sum_{i=1}^d \left(\frac{x_k^i - x_j^i}{r_i} \right)^2} \right) \tag{2}$$

where P_k is the potential of k th data point and r_i is the cluster radius associated with i th dimension of the point. After determining the potential value of each data point, the point with the highest potential is selected as the first cluster center. Assume x_{c1} is that point and P_1^* is its potential value. Then, the potential value of each data point x_k is reduced by using the following equation:

$$P'_k = P_k - P_1^* \exp \left(-4 \sqrt{\sum_{i=1}^d \left(\frac{x_k^i - x_{c1}^i}{\eta r_i} \right)^2} \right) \tag{3}$$

where P'_k is the reduced potential value of the k th data point and η is a parameter called as squash factor, which is multiplied by radius values to determine the neighboring clusters within which the existence of other cluster centers are discouraged. The acceptance level of data points with potential values between the upper and lower thresholds depends on the relative distance equation as follows:

$$d_{\min} + \frac{P_k^*}{P_1^*} \geq 1 \tag{4}$$

where d_{\min} is the nearest distance between the candidate cluster center.

2.3 Integrated ANFIS and Data Mining

The integrated ANFIS and data mining model proposed in this study are optimized using a Genetic algorithm (GA). So, the mapping input values to desired outputs are optimized by GA-ANFIS method proposed to get the total prediction error of the final model minimized.

3 Experimental Results

The Ni-MH battery used in the investigation is of nominal 85 Ah Ni-MH modules, giving a nominal voltage of 12V.

The main requirement for fast charging is minimal charging time without abuse to the battery. Generally, we first discharged the module at C/3 rate to the desired state of charge (SOC), and then recharged it in different charge regimes.

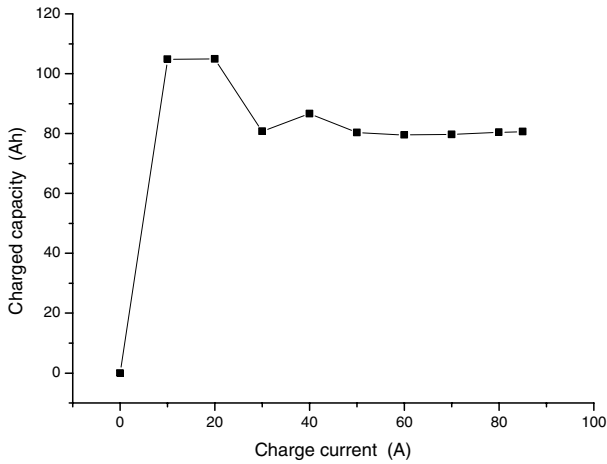


Fig. 2. Correlation between charge current and charged capacity

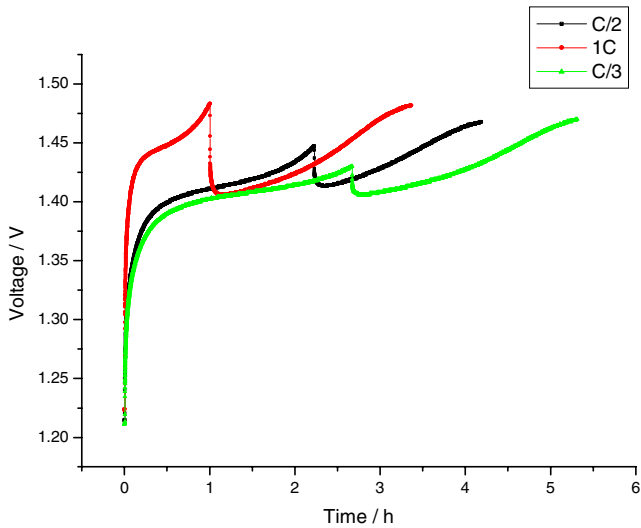


Fig. 3. Voltage vs. charge time in various two-step constant current charging regimes

The charge test protocols included constant current (CC) and multistage constant current (MCC) charging regimes and following the USABC Electric Vehicle Battery Test Procedures Manual [10] — fast charge tests procedure.

In all charge tests, the module voltage, temperature and charge input were monitored. In CC charge, we can apply different charging currents as control input with values ranging from 0A to 85A. Fig.2 shows the correlation between charged capacity and charge current. The charged capacity and charge current profiles depict the optimum

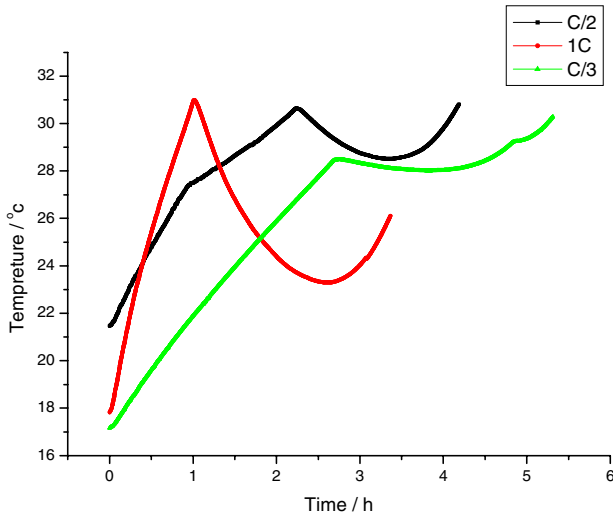


Fig. 4. Time vs. charge time in various two-step constant current charging regimes

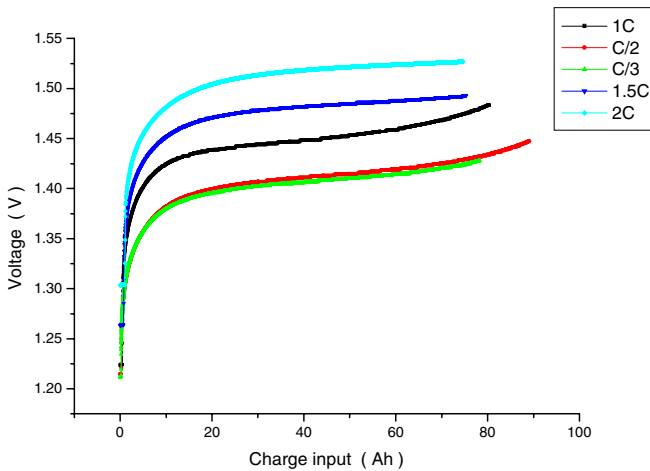


Fig. 5. Voltage vs. charge input in various constant current charging regimes

charge current is about 0.1C in CC charge regimes. MCC charging consists of two CC steps, which starts with a high CC charging for a certain time, followed by a low cc charging. The comparison of voltage and temperature are shown in Fig.3 and Fig.4. The correlation of voltage and charge input is shown in Fig.5 in fast charging tests.

As it is mentioned above, there is no available formal mathematical model of the battery under the charging process. We propose the integrated ANFIS and dynamic data mining model can most efficiently be managed. In the fast charge tests, we found that the primary controlling parameters are the voltage and temperature. Therefore, the input signals, voltage(ΔV), differential of temperature (dT/dt), are fuzzified to respective fuzzy terms used in the rules. The controller performs fuzzy inference and determines fuzzy values of control signal. The defuzzified fuzzy control signal, representing the value of current (I) is then applied to the battery. The profile of single input variable vs. output variable and the input vs. output view surface are shown in Fig.6 and Fig.7, respectively. Fig.8 shows the training error.

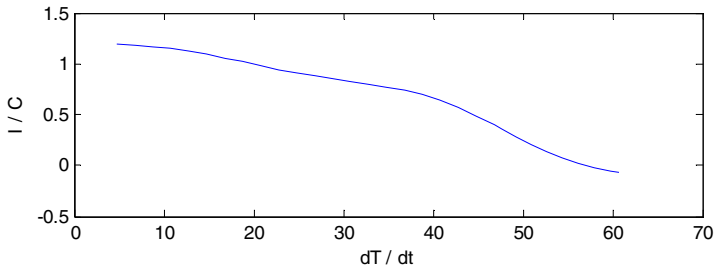


Fig. 6. Correlation between charge current and dT/dt ($\Delta V=Med$)

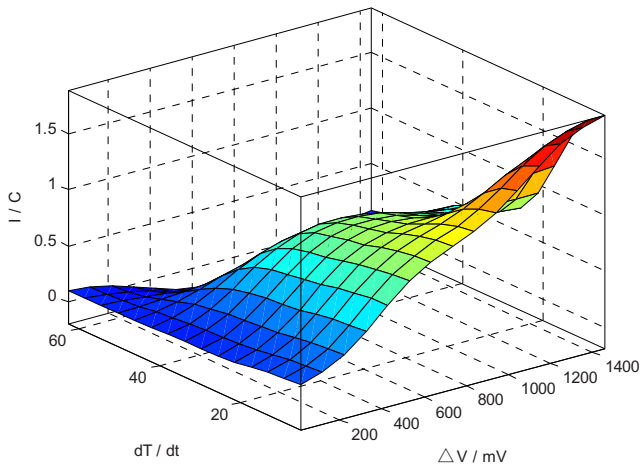


Fig. 7. Correlation view surface n Input vs. output variables

The discussion above suggests that a optimal charging control for Ni-MH should use relatively low current in the beginning and near the end of charging. Moreover,

If ΔV is High and dT/dt is Low, then $I=1.5C$, namely, start with the fast charging;
If ΔV is Low and dT/dt is Low, then $I=0.1C$, namely, end with fast charging.

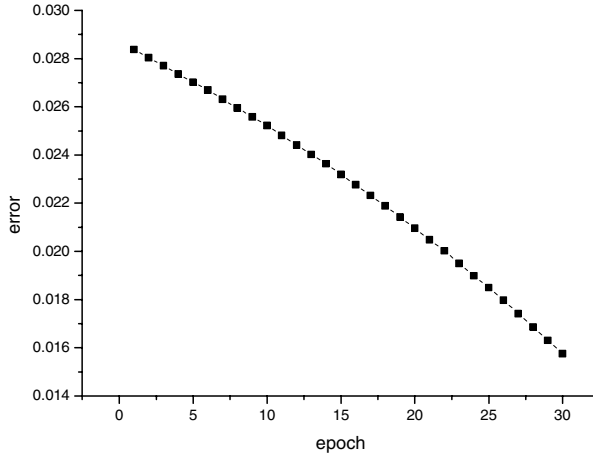


Fig. 8. Training error

4 Conclusion

In this study we use the dynamic data mining technique on the basis ANFIS to extract fuzzy rules for the Ni-MH battery fast intelligent charging controller. The results show that the suggested approach gives charging efficiency and avoided overcharge and overheating. Moreover, a low charging rate be desired for charging the initial for 10% capacity. This proposed approach which can be widely used for unknown systems and creating self-adaptable intelligent controllers is general and can be extended to design controllers for quickly charging different battery types.

References

1. Castillo, O., Melin, P.: *Soft Computing for Control of Non-Linear Dynamical Systems*. Springer, Germany (2001)
2. Ionescu, P.D., Moscalu, M., Mosclu, A.: Intelligent Charger with Fuzzy Logic. In: *Int. Symp. on Signals, Circuits and Systems 2003(SCS 2003)* (2003)
3. Khosla, A., Kumar, S., Aggarwal, K.K.: Fuzzy Controller for Rapid Nickel-cadmium Batteries Charger through Adaptive Neuro-fuzzy Inference System(ANFIS) Architecture. In: *22nd International Conference of the North American Fuzzy Information Processing Society, NAFIPS 2003*, pp. 534–544 (2003)
4. Yang, X.G., Liaw, B.Y.: Fast Charging Nickel-metal Hydride Traction Batteries. *J. Power Sources* 101, 158–166 (2001)
5. Aliev, R.A., Aliev, R.R., Guirimov, B., Uyar, K.: Dynamic Data Mining Technique for Rules Extraction in a Process of Battery Charging. *J. Applied Computing* 8, 1252–1258 (2008)

6. Fayyad, U., Piatetsky-Shapiro, G., Smith, P.: From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, 37–53 (1996)
7. Han, J., Kamber, M.: *Data Mining: Concepts and Techniques*. Morgan Kaufmann, San Francisco (2001)
8. Huang, M.J., Chen, M.Y., Lee, S.C.: Integrating Data Mining with Case-based Reasoning for Chronic Diseases Prognosis and Diagnosis. *Expert Systems with Applications* 32, 856–867 (2007)
9. Chiu, S.L.: Fuzzy Model Identification Based on Cluster Estimation. *Journal of Intelligent Fuzzy System*, 267–278 (1994)
10. IDAHO National Engineering Lab.: *USABC Electric Vehicle Battery Test Procedures Manual, Revision 2, DOE/ID-10479* (1996)

Robust and Adaptive Fuzzy Feedback Linearization Regulator Design

Chang-Woo Park¹, Young-Ouk Kim², and Chong Ho Yi¹

¹ Dept. of Digital & information, INHA Technical College
253, Yonghyun-Dong, Nam-gu, Incheon, Korea
{drcwpark, ycm}@inhatec.ac.kr

² Intelligent Robotics Research Center, Korea Electronics Technology Institute
kimyo@keti.re.kr

Abstract. In this paper, we have proposed a new adaptive fuzzy control algorithm and the regulation problem based on Takagi-Sugeno fuzzy model. The regulation problem for the uncertain SISO nonlinear system is solved by the proposed algorithm. Using the advanced stability theory, the stability of the state, the control gain and the parameter choosing considering the approximation error due to the fuzzy modeling is proved. Some numerical examples are presented to verify the effectiveness of the proposed methods.

1 Introduction

Since Mamdani has applied fuzzy control to a steam engine, many successful applications have been reported [1]-[4]. In the early stage of fuzzy control, fuzzy control was a direct method for controlling a system without the need of a mathematical model, in contrast to the classical control which is an indirect method with a mathematical model. However, the stability analysis of the indirect or 'model-free' method is difficult and apt to yield a very conservative result. For this reason, the latest fuzzy control researches have been focused on the indirect fuzzy control method. In the indirect fuzzy control method, the fuzzy system is used to estimate and approximate the plant dynamics. For mathematical simplicity of analysis, singleton fuzzy system or Takagi-Sugeno fuzzy model is frequently used [5]. Theoretically, the fuzzy system can exactly approximate the dynamics of a class of nonlinear plants with the large or infinite number of rules. However, in the practical situations of limited number of rules and incorrect approximation, uncertainties are inevitably produced. To overcome the effect of uncertainties on stability, the various adaptive fuzzy control algorithms are proposed [6]-[15]. Most of the existing adaptive fuzzy control algorithms are based on the feedback linearization method [9]-[15]. However, the feedback linearization method can not be applied to the plant with the singularity in the inverse dynamics. The adaptive fuzzy control algorithms based on the feedback linearization method need the infinite control input, when the state is at singularity of the inverse dynamics or the parameter approximation error diverges to infinity. To avoid the need of the infinite control input, the existing feedback linearization based methods require the assumption that the state is away from the singularity and the projection algorithm which prevents the parameter approximation error from diverging to infinity.

In this paper, we have proposed the indirect adaptive fuzzy state feedback regulator using Takagi-Sugeno fuzzy model. The proposed method is less sensitive to singularity than the adaptive fuzzy control algorithms based on the feedback linearization method. And it can guarantee the boundedness of the state and the parameter approximation error directly in the face of the bounded approximation error and external disturbance and we have also proved that the system state converges to equilibrium.

2 Problem Formulation

Consider the regulation problem of the following n -th order nonlinear SISO system.

$$x^{(n)} = f(\mathbf{x}) + g(\mathbf{x})u + d(t) \tag{1}$$

where f and g are unknown (uncertain) but bounded continuous nonlinear functions and $d(t)$ denotes the external disturbance which is unknown but bounded in magnitude. Also, u denotes the control input. Let $\mathbf{x} = [x, \dot{x}, \dots, x^{(n-1)}]^T \in R^n$ be the state vector of the system which is assumed to be available.

In this paper, well-known Takagi-Sugeno fuzzy model is used to identify the unknown nonlinear system (1). Takagi-Sugeno fuzzy model is available in IF-THEN form (2) or Input-Output form (3).

plant rule i :

$$\begin{aligned} \text{IF } x \text{ is } M_{i1} \text{ and } \dot{x} \text{ is } M_{i2} \text{ and } \dots \text{ and } x^{(n-1)} \text{ is } M_{in} \\ \text{THEN } x^{(n)} = \mathbf{a}_i^T \mathbf{x} + b_i u, \quad i = 1, 2, \dots, r \end{aligned} \tag{2}$$

$$x^{(n)} = \frac{\sum_{i=1}^r w_i(\mathbf{x}) \{ \mathbf{a}_i^T \mathbf{x} + b_i u \}}{\sum_{i=1}^r w_i(\mathbf{x})} = \sum_{i=1}^r h_i(\mathbf{x}) \{ \mathbf{a}_i^T \mathbf{x} + b_i u \} \tag{3}$$

By defining the parameter vectors $\boldsymbol{\theta}_a$ and $\boldsymbol{\theta}_b$ as

$$\boldsymbol{\theta}_a = [\mathbf{a}_1^T \ \mathbf{a}_2^T \ \dots \ \mathbf{a}_r^T]^T \in R^{nr} \text{ and } \boldsymbol{\theta}_b = [b_1 \ b_2 \ \dots \ b_r]^T \in R^r,$$

Input-Output form (3) can be expressed by the following equivalent equations.

$$x^{(n)} = \hat{f}(\mathbf{x}|\boldsymbol{\theta}_a) + \hat{g}(\mathbf{x}|\boldsymbol{\theta}_b)u + d(t) \tag{4-1}$$

$$\hat{f}(\mathbf{x}|\boldsymbol{\theta}_a) = \boldsymbol{\theta}_a^T \boldsymbol{\xi}_a = \sum_{i=1}^r h_i(\mathbf{x}) \mathbf{a}_i^T \mathbf{x} \tag{4-2}$$

$$\hat{g}(\mathbf{x}|\boldsymbol{\theta}_b) = \boldsymbol{\theta}_b^T \boldsymbol{\xi}_b = \sum_{i=1}^r h_i(\mathbf{x}) b_i u \tag{4-3}$$

where,

$$\boldsymbol{\xi}_a = [h_1(\mathbf{x})\mathbf{x}^T \ h_2(\mathbf{x})\mathbf{x}^T \ \dots \ h_r(\mathbf{x})\mathbf{x}^T]^T \in R^{nr}, \ \boldsymbol{\xi}_b = [h_1(\mathbf{x}) \ h_2(\mathbf{x}) \ \dots \ h_r(\mathbf{x})]^T \in R^r$$

3 Indirect Adaptive Fuzzy State Feedback Regulator

First, we define the optimal parameter vectors as

$$\boldsymbol{\theta}_a^* = \arg \min_{\boldsymbol{\theta}_a \in R^{nr}} \left[\sup_{\mathbf{x} \in R^n} \left| \hat{f}(\mathbf{x} | \boldsymbol{\theta}_a) - f(\mathbf{x}) \right| \right] \quad (5-1)$$

$$\boldsymbol{\theta}_b^* = \arg \min_{\boldsymbol{\theta}_b \in R^r} \left[\sup_{\mathbf{x} \in R^n} \left| \hat{g}(\mathbf{x} | \boldsymbol{\theta}_b) - g(\mathbf{x}) \right| \right] \quad (5-2)$$

where, $\boldsymbol{\theta}_a^* = [\mathbf{a}_1^{*T} \ \mathbf{a}_2^{*T} \ \cdots \ \mathbf{a}_r^{*T}]^T \in R^{nr}$, $\boldsymbol{\theta}_b^* = [b_1^* \ b_2^* \ \cdots \ b_r^*]^T \in R^r$

Thus $\hat{f}(\mathbf{x} | \boldsymbol{\theta}_a^*)$ and $\hat{g}(\mathbf{x} | \boldsymbol{\theta}_b^*)$ are the optimal approximators of $f(\mathbf{x})$ and $g(\mathbf{x})$, respectively, among all the fuzzy systems in the form of (4-2) and (4-3). Considering uncertainties, the uncertain system (1) can be represented by the optimal approximations as in the following equations.

$$\dot{x}^{(n)} = f(\mathbf{x}) + g(\mathbf{x})u + d(t) \quad (6-1)$$

$$f(\mathbf{x}) = \hat{f}(\mathbf{x} | \boldsymbol{\theta}_a^*) + \Delta \mathbf{a}(t)^T \mathbf{x} = \sum_{i=1}^r h_i(\mathbf{x}) \mathbf{a}_i^{*T} \mathbf{x} + \Delta \mathbf{a}(t)^T \mathbf{x} \quad (6-2)$$

$$g(\mathbf{x}) = \hat{g}(\mathbf{x} | \boldsymbol{\theta}_b^*) + \Delta b(t) = \sum_{i=1}^r h_i(\mathbf{x}) b_i^* u + \Delta b(t) \quad (6-3)$$

where $\Delta \mathbf{a}(t) \in R^n$ and $\Delta b(t) \in R$ denote the time-varying uncertainties which are assumed to be bounded by some known $\Delta \mathbf{a}$ and Δb as follows.

$$\Delta \mathbf{a} = [\Delta a_1 \ \Delta a_2 \ \cdots \ \Delta a_n]^T \in R^{n+}, \quad |\Delta a_j(t)| \leq \Delta a_j \quad , \text{ for all } j$$

$$\Delta b \in R^+, \quad |\Delta b(t)| \leq \Delta b$$

Also, the external disturbance is assumed to be bounded by some known d .

$$d \in R^+$$

$$|d(t)| \leq d$$

To regulate the uncertain system (6-1) - (6-3), we propose the following indirect adaptive fuzzy state feedback regulator (7-1) - (7-4).

$$u = \mathbf{k}^T \mathbf{x} \quad (7-1)$$

$$\dot{\mathbf{k}} = -\Lambda \hat{\Gamma}(\mathbf{x}, \mathbf{k}, \tilde{\mathbf{a}}_i, \tilde{b}_i, t) \mathbf{P} \mathbf{x} \quad (7-2)$$

$$\dot{\mathbf{a}}_i^T = \alpha_1 h_i(\mathbf{x}) \mathbf{x}^T \mathbf{P} \mathbf{w} \mathbf{x}^T \quad (7-3)$$

$$b_i = \alpha_2 h_i(\mathbf{x}) \mathbf{x}^T \mathbf{P} \mathbf{w} \mathbf{k}^T \mathbf{x} \quad (7-4)$$

where,

i) $\tilde{\mathbf{a}}_i$ and \tilde{b}_i are the parameter approximation errors defined as $\tilde{\mathbf{a}}_i = \mathbf{a}_i^* - \mathbf{a}_i$, $\tilde{b}_i = b_i^* - b_i$ for all i , respectively.

ii) $\mathbf{k} \in R^n$ is the adaptive state feedback gain vector. $\mathbf{\Lambda}$ is any $n \times n$ symmetric positive definite adaptation gain matrix. α_1 and α_2 are the positive adaptation gain constants.

iii) \mathbf{P} is the $n \times n$ symmetric positive definite solution of the Lyapunov equation $\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} = -\mathbf{Q}$, with \mathbf{Q} any symmetric positive definite matrix.

In this paper, \mathbf{A} is chosen to satisfy that

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ \underline{a}_1 & \underline{a}_2 & \underline{a}_3 & \cdots & \underline{a}_n \end{bmatrix} \text{ is any Hurwitz matrix, } \mathbf{A} \in R^{n \times n}$$

iv) $\mathbf{w} = [0 \ \cdots \ 0 \ 1]^T \in R^n$.

v) $\hat{\mathbf{\Gamma}}(\mathbf{x}, \mathbf{k}, \tilde{\mathbf{a}}_i, \tilde{b}_i, t)$ is a $n \times n$ matrix which can be computed as (8-1) and (8-2).

$$\hat{\mathbf{\Gamma}}(\mathbf{x}, \mathbf{k}, \tilde{\mathbf{a}}_i, \tilde{b}_i, t) = \hat{\mathbf{r}} \mathbf{w}^T, \quad \hat{\mathbf{r}} \in R^n \tag{8-1}$$

$$\begin{aligned} \hat{\mathbf{r}} &= \sum_{i=1}^r h_i(\mathbf{x}) \left\{ \frac{\mathbf{k} \mathbf{x}^T (\mathbf{a}_i + \mathbf{R} \Delta \mathbf{a} - \mathbf{a})}{\mathbf{k}^T \mathbf{k}} + (b_i + \Delta b \mathbf{S}) \mathbf{x} \right\} + \frac{\text{sgn}(x_n) d \mathbf{k}}{\mathbf{k}^T \mathbf{k}} \\ &= \sum_{i=1}^r h_i(\mathbf{x}) \left\{ \frac{\mathbf{k} \mathbf{x}^T (\mathbf{a}_i^* - \tilde{\mathbf{a}}_i + \mathbf{R} \Delta \mathbf{a} - \mathbf{a})}{\mathbf{k}^T \mathbf{k}} + (b_i^* - \tilde{b}_i + \Delta b \mathbf{S}) \mathbf{x} \right\} + \frac{\text{sgn}(x_n) d \mathbf{k}}{\mathbf{k}^T \mathbf{k}} \end{aligned} \tag{8-2}$$

In (8-2), \mathbf{a} is the transpose vector of the last row of \mathbf{A} . That is,

$$\mathbf{a} = [\underline{a}_1 \ \underline{a}_2 \ \underline{a}_3 \ \cdots \ \underline{a}_n]^T \in R^n.$$

Also, \mathbf{R} and \mathbf{S} are the diagonal matrixes defined as follows.

$$\mathbf{R} \equiv \begin{bmatrix} \text{sgn}(x_1 x_n) & 0 & \cdots & 0 & 0 \\ 0 & \text{sgn}(x_2 x_n) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \text{sgn}(x_{n-1} x_n) & 0 \\ 0 & 0 & \cdots & 0 & \text{sgn}(x_n x_n) \end{bmatrix} \tag{9-1}$$

$$\mathbf{S} \equiv \begin{bmatrix} \text{sgn}(k_1 x_1 x_n) & 0 & \cdots & 0 & 0 \\ 0 & \text{sgn}(k_2 x_2 x_n) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \text{sgn}(k_{n-1} x_{n-1} x_n) & 0 \\ 0 & 0 & \cdots & 0 & \text{sgn}(k_n x_n x_n) \end{bmatrix} \tag{9-2}$$

sgn in (9-1) and (9-2) is the sign function:

Substituted in (6-1) - (6-3), the adaptive control laws (7-1) - (7-4) give the following closed loop dynamic equation.

$$\dot{\mathbf{x}} = \underline{\mathbf{A}}\mathbf{x} + \Gamma^T(\mathbf{x}, \mathbf{k}, t)\mathbf{k} \tag{10-1}$$

$$\dot{\mathbf{k}} = -\Lambda \hat{\Gamma}(\mathbf{x}, \mathbf{k}, \tilde{\mathbf{a}}_i, \tilde{b}_i, t) \mathbf{P} \mathbf{x} \tag{10-2}$$

$$\dot{\tilde{\mathbf{a}}}_i^T = -\alpha_1 h_i(\mathbf{x}) \mathbf{x}^T \mathbf{P} \mathbf{w} \mathbf{x}^T \tag{10-3}$$

$$\dot{\tilde{b}}_i = -\alpha_2 h_i(\mathbf{x}) \mathbf{x}^T \mathbf{P} \mathbf{w} \mathbf{k}^T \mathbf{x} \tag{10-4}$$

Where $\dot{\tilde{\mathbf{a}}}_i = -\dot{\mathbf{a}}_i, \dot{\tilde{b}}_i = -\dot{b}_i$.

$\Gamma(\mathbf{x}, \mathbf{k}, t)$ is a $n \times n$ matrix which can be computed as (11-1) and (11-2).

$$\Gamma(\mathbf{x}, \mathbf{k}, t) = \mathbf{r} \mathbf{w}^T, \quad \mathbf{r} \in R^n \tag{11-1}$$

$$\mathbf{r} = \sum_{i=1}^r h_i(\mathbf{x}) \left\{ \frac{\mathbf{k} \mathbf{x}^T (\mathbf{a}_i^* + \Delta \mathbf{a}(t) - \underline{\mathbf{a}})}{\mathbf{k}^T \mathbf{k}} + (b_i^* + \Delta b(t)) \mathbf{x} \right\} + \frac{d(t) \mathbf{k}}{\mathbf{k}^T \mathbf{k}} \tag{11-2}$$

Theorem 1: For the above closed loop dynamic equation (10-1) - (10-4),

$$\text{If } \mathbf{P} \mathbf{w} = p_{nm} \mathbf{w}, \quad p_{nm} > 0 \tag{12}$$

is satisfied, then the following statements hold.

(a) the equilibrium point $(\mathbf{x}, \mathbf{k}, \tilde{\mathbf{a}}_i, \tilde{b}_i) = \mathbf{0}$ is uniformly stable.

(b) $\mathbf{x}, \mathbf{k}, \tilde{\mathbf{a}}_i, \tilde{b}_i$ are uniformly bounded $\forall t \geq t_0, \forall \mathbf{x}(t_0) \in R^n, \forall \mathbf{k}(t_0) \in R^p,$

$\forall \tilde{\mathbf{a}}_i(t_0) \in R^n, \forall \tilde{b}_i(t_0) \in R.$

(c) $\lim_{t \rightarrow \infty} \|\mathbf{x}(t)\| = 0, \forall \mathbf{x}(t_0) \in R^n, \forall \mathbf{k}(t_0) \in R^p, \forall \tilde{\mathbf{a}}_i(t_0) \in R^n, \forall \tilde{b}_i(t_0) \in R$

From **Theorem 1**, we can conclude that the proposed adaptive fuzzy control algorithm can regulate the uncertain nonlinear system with the bounded control input and the parameter approximation error.

4 Example

Consider the problem of balancing and swing-up of an inverted pendulum on a cart. The equations of motion for the pendulum are [17].

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(\mathbf{x}) + g(\mathbf{x}) + d(t) = \frac{g \sin(x_1) - a m l x_2^2 (2x_1) / 2 - a \cos(x_1) u}{4l / 3 - a m l \cos^2(x_1)} + d(t) \end{aligned} \tag{13}$$

where $\mathbf{x} = [x_1 \quad x_2]^T$; x_1 denotes the angle (in radians) of the pendulum from the vertical, and x_2 is the angular velocity. $g = 9.8 m / s^2$ is the gravity constant, m is the mass of the pendulum, M is the mass of the cart, $2l$ is the length of the pendulum,

u is the control force applied to the cart (in Newtons). $d(t)$ is the external disturbance and $a = \frac{1}{m + M}$. We choose $m = 2.0\text{kg}$, $M = 8.0\text{kg}$, $2l = 1.0\text{m}$ in the simulation. For the convenience of simulation, the external disturbance $d(t)$ is assumed to be a square wave with the amplitude ± 0.05 and the period 2π .

As a model for the pendulum, we use the following Takagi-Sugeno fuzzy model with two rules. Membership functions are shown in Fig. 1.

Rule1: IF x is about 0

THEN $\dot{x}_2 = \mathbf{a}_1^T \cdot \mathbf{x} + b_1 u$

Rule2: IF x is about $\pm \frac{\pi}{2}$ ($|x| < \frac{\pi}{2}$)

THEN $\dot{x}_2 = \mathbf{a}_2^T \cdot \mathbf{x} + b_2 u$

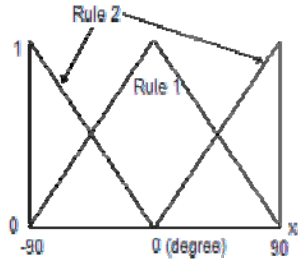


Fig. 1. Membership function

In this simulation, the time-varying approximation errors are assumed to be bounded as follows

$$\Delta \mathbf{a} = [\Delta a_1 \quad \Delta a_2]^T = [3 \quad 0]^T \in R^{2+}, \quad |\Delta a_j(t)| \leq \Delta a_j, \quad j = 1, 2$$

$$\Delta b = 0.1 \in R^+, \quad |\Delta b(t)| \leq \Delta b \tag{14}$$

To show the validity of the above assumption, we suppose that $f(\mathbf{x})$ and $g(\mathbf{x})$ are known and $d(t) = 0$ in the system. Then, applying the optimization technique in (5-1) and (5-2), the optimal parameter vectors can be obtained as

$$\boldsymbol{\theta}_a^* = [\mathbf{a}_1^{*T} \quad \mathbf{a}_2^{*T}]^T \in R^4, \quad \boldsymbol{\theta}_b^* = [b_1^* \quad b_2^*]^T \in R^2$$

where, $\mathbf{a}_1^* = [17.29 \quad 0]^T$, $\mathbf{a}_2^* = [9.35 \quad 0]^T$, $b_1^* = -0.18$, $b_2^* = -0.01$.

With these results, since $f(\mathbf{x}) - \hat{f}(\mathbf{x}|\boldsymbol{\theta}_a^*) = \Delta \mathbf{a}(t)^T \mathbf{x}$ and $g(\mathbf{x}) - \hat{g}(\mathbf{x}|\boldsymbol{\theta}_b^*) = \Delta b(t)$ from (6-1) and (6-2), it can be easily verified that the time varying uncertainties are bounded. For the system, computer simulation is performed by applying the

proposed algorithm. The conditions used in the computer simulation are summarized as follows.

$$\begin{aligned} \mathbf{x}(t_0)^T &= [2 \ 0] \ , \ \mathbf{k}(t_0)^T = [-25 \ -25] \ , \\ \mathbf{a}_1 &= [15 \ 0]^T \ , \ \mathbf{a}_2 = [10 \ 0]^T \ , \\ b_1 &= -0.1 \ , \ b_2 = -0.01 \ , \\ \mathbf{\Lambda} &= \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix} \ , \ \mathbf{A} = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix} \ , \ \mathbf{P} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ \alpha_1 &= \alpha_2 = 0.1 \ . \end{aligned}$$

Fig. 2. (a) and (b) show the simulation results of the state variable x_1 and the control input u . From these figures, we can conclude that the proposed algorithm can achieve a regulation problems for the uncertain nonlinear system with the bounded control input and the parameter approximation error successfully.

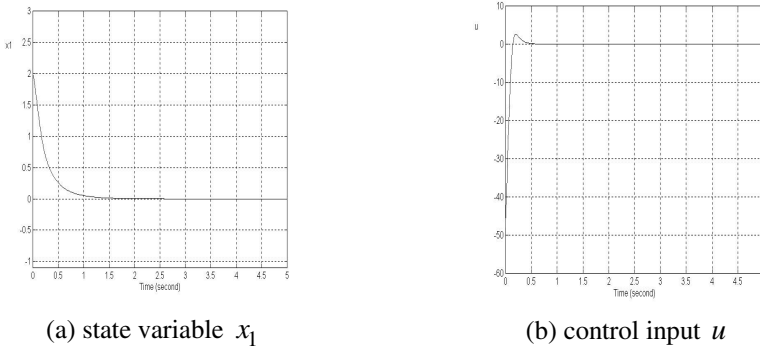


Fig. 2. Simulation results

5 Conclusion

A new adaptive fuzzy control method is proposed to regulate the uncertain SISO nonlinear system. The proposed method has the dynamic state feedback structure with the adaptive fuzzy system identification. The proposed method can guarantee not only the local stability but also the global stability. Also, the proposed method can guarantee the boundedness of both the control input and the parameter approximation error directly without the additional projection algorithm.

References

1. Sugeno, M.: Industrial Applications of Fuzzy Control. Elsevier Science, Amsterdam (1985)
2. Lee, C.C.: Fuzzy Logic in Control: Fuzzy Logic Controller-Part I and Part II. IEEE Trans. Syst., Man, Cybern. 20, 404–435 (1990)

3. Kosko, B.: *Neural Networks and Fuzzy Systems*. Prentice-Hall, Englewood Cliffs (1992)
4. Jamshidi, M., Vadiiee, N., Ress, T.J.: *Fuzzy Logic and Control*. Prentice-Hall, Englewood Cliffs (1993)
5. Takagi, T., Sugeno, M.: Fuzzy Identification of Systems and Its Application to Modeling and Control. *IEEE Trans. Syst., Man, Cybern.* 15, 116–132 (1985)
6. Kong, S.G., Kosko, B.: Adaptive Fuzzy Systems for Backing up a Truck and Trailer. *IEEE Trans. Neural Networks* 3, 211–223 (1992)
7. Su, C.Y., Stepanenko, Y.: Adaptive Control of a Class of Nonlinear Systems with Fuzzy Logic. *IEEE Trans. Fuzzy Syst.* 2, 285–294 (1994)
8. Wong, C.C., Huang, B.C., Chen, J.Y.: Rule Regulation of Indirect Adaptive Fuzzy Controller Design. *IEE Proc. Control Theory Appli.* 145, 513–518 (1988)
9. Wang, L.X.: Stable Adaptive Fuzzy Control of Nonlinear Systems. *IEEE Trans. Fuzzy Syst.* 1, 146–155 (1993)
10. Wang, L.X.: *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*. Prentice-Hall, Englewood Cliffs (1993)
11. Spooner, J.T., Passino, K.M.: Stable Adaptive Control Using Fuzzy Systems and Neural Networks. *IEEE Trans. Fuzzy Syst.* 4, 339–359 (1996)
12. Chen, B.S., Lee, C.H., Chang, Y.C.: H Graphics object to be inserted manually Tracking Design of Uncertain Nonlinear SISO Systems: Adaptive Fuzzy Approach. *IEEE Trans. Fuzzy Syst.* 4, 32–43 (1996)
13. Fischle, K., Schroder, D.: An Improved Stable Adaptive Fuzzy Control Method. *IEEE Trans. Fuzzy Syst.* 7, 27–40 (1999)
14. Tsay, D.L., Chung, H.Y., Lee, C.J.: The Adaptive Control of Nonlinear Systems Using the Sugeno-type of Fuzzy Logic. *IEEE Trans. Fuzzy Syst.* 7, 225–229 (1999)
15. Ordonez, R., Passino, K.M.: Stable Multi-input Multi-output Adaptive Fuzzy/Neural Control. *IEEE Trans. Fuzzy Syst.* 7, 345–353 (1999)
16. Sastry, S.S., Isidori, A.: Adaptive Control of Linearizable Systems. *IEEE Trans. Automat. Contr.* 34, 1123–1131 (1989)
17. Cannon, R.H.: *Dynamics of Physical Systems*. McGrawHill, New York (1967)
18. Sun, F.C., Sun, Z.Q., Feng, G.: An Adaptive Fuzzy Controller Based on Sliding Mode for Robot Manipulators. *IEEE Trans. Syst., Man, Cybern.* 29, 661–667 (1999)
19. Leu, Y.G., Lee, T.T., Wang, W.Y.: Observer Based Adaptive Fuzzy-neural Control for Unknown Nonlinear Dynamical Systems. *IEEE Trans. Syst., Man, Cybern.* 29, 583–591 (1999)
20. Sanner, R.M., Slotine, J.J.E.: Gaussian Networks for Direct Adaptive Control. *IEEE Trans. Neural Networks* 3, 837–863 (1992)
21. Narendra, K.S., Annaswamy, A.M.: *Stable Adaptive Systems*. Prentice-Hall, Englewood Cliffs (1989)
22. Rovithakis, G.A., Christodoulou, M.A.: Adaptive Control of Unknown Plants Using Dynamic Neural Networks. *IEEE Trans. Syst., Man, Cybern.* 24, 400–412 (1994)
23. Ioannou, P.A., Sun, J.: *Robust Adaptive Control*. Prentice Hall, New Jersey (1996)

Three-Level Image Segmentation Based on Maximum Fuzzy Partition Entropy of 2-D Histogram and Quantum Genetic Algorithm

Hai-Yan Yu and Jiu-Lun Fan

Department of Information and Control, Xi'an Institute of Post and Telecommunications,
Xi'an 710121, China
yuhaiyan2010@yahoo.com.cn

Abstract. A method is presented for three-level image segmentation through maximizing the fuzzy partition entropy of two-dimensional histogram. Two groups, each including three member functions, namely Z-function, Π -function and S-function, are used for fuzzy division of two-dimensional histogram to get nine fuzzy sets. And the nine fuzzy sets are classified to three parts, corresponding to dark, gray and white part of the image, respectively, while a fuzzy partition is obtained for the two-dimensional space. Then a fuzzy partition entropy is defined based on multi-dimensional fuzzy partition and entropy theory. The parameters of the six membership functions can be determined by maximizing fuzzy partition entropy of two-dimensional histogram and the procedure for finding the optimal combination of all the fuzzy parameters is implemented by quantum genetic algorithm with an appropriate coding method. The experiment results show that the proposed method gives better performance than one-dimensional three-level thresholding method under noise case.

Keywords: Image segmentation; Fuzzy partition; Quantum genetic algorithm.

1 Introduction

The goal of image segmentation is to extract meaningful objects from an input image. Image segmentation, with wide recognized significance, is one of the most difficult low-level image analysis tasks. All the subsequent tasks, including feature extraction, model matching and object recognition rely heavily on the quality of the image segmentation process. Using fuzzy set to image segmentation is a main way. There have been numerous applications of fuzzy entropies in image segmentation. Cheng *et al.* [1] proposed fuzzy homogeneity vectors to handle the grayness and spatial uncertainties among pixels, and to perform multilevel thresholding. Zhao *et al.* [2] presented an entropy function by the fuzzy c-partition (FP) and the probability partition (PP), and applied to three-level image segmentation. Tao *et al.* [3] also showed a 1-D three-level thresholding method through maximizing the fuzzy entropy with the fuzzy region represented by three member functions of Z-function and Π -function and S-function, which got better performances than Zhao's method. Jing *et al.* [4] gave a

method for image segmentation through maximizing the fuzzy partition entropy of 2-D histogram, which presented herein performs better than some classical threshold selection methods for two-level image segmentation.

In this paper, a three-level image segmentation method is put forward based on fuzzy partition of 2-D histogram combined with multi-dimensional fuzzy partition entropy. Two groups, each including three member functions, namely Z -function, Π -function and S -function, are used for fuzzy division of 2-D histogram to get nine fuzzy subsets. Then the nine fuzzy subsets are classified to three parts, corresponding to dark, gray and white part of the image, respectively, while a fuzzy partition is obtained for a sample space. Then, a new fuzzy partition entropy of 2-D histogram is presented. The parameters of the six membership functions can be determined by maximizing the fuzzy partition entropy. For finding the optimal combination of all the fuzzy parameters, the choice of optimization algorithm is an important step to solve the problem. Quantum genetic algorithm (QGA) [5] is the product of quantum computing combined with genetic algorithm (GA) [6], which adopts the multi-state gene qubit coding and quantum rotation gate strategy irrelative to the tasks, utilizes dynamic adjusting rotation angle mechanism, making the algorithm more adaptable and efficient. Therefore, we use QGA to automatically select an optimal combination of the parameters with fuzzy partition entropy as fitness function. Experiments show that our method can obtain better segmentation results than Tao’s method.

2 Multi-dimensional Fuzzy Partition Entropy

2.1 Fuzzy Partition Entropy

Let (Ω, E, p) be a probability space in which Ω is the sample space. $E \subset P(\Omega)$ is the σ -field of Borel sets in Ω and $p: E \rightarrow [0,1]$ is a probability measure over Ω . Let $\tilde{A} \in F(\Omega)$ be a fuzzy set in (Ω, E, p) , whose membership function is μ_A ($\mu_A: E \rightarrow [0,1]$). The probability of a fuzzy event \tilde{A} is defined by $p(\tilde{A}) = \int_{\Omega} \tilde{A}(\omega) dp$. Let \tilde{A}, \tilde{B} be fuzzy sets in probability space (Ω, E, p) , the conditional probability of \tilde{A} given \tilde{B} is: $p(\tilde{A} | \tilde{B}) = p(\tilde{A}\tilde{B}) / p(\tilde{B})$ [4], [7].

Let $X = \{X_1, X_2, \dots, X_n\} \subset R^p$. $X_k = (X_{k1}, X_{k2}, \dots, X_{kn})^T \subset R^p$ are eigenvectors, where X_{kj} is the j -th eigenvector. X is divided into c fuzzy sets \tilde{C}_i , which is a fuzzy partition of $X = \{X_1, X_2, \dots, X_n\} \subset R^p$, where $\sum_{i=1}^c \mu_{ik} = 1, \forall X_k \in X, 2 \leq c \leq n$ and $\mu_{ik} = \mu_i(X_k) \in [0,1]$ is the membership function of \tilde{C}_i [8]. Let $P = \{\tilde{A}_1, \tilde{A}_2, \dots, \tilde{A}_c\}$ be a fuzzy partition on $X \subset R^p$, \tilde{B} be an fuzzy event in the probability space (X, E, p) . The conditional probability of fuzzy partition P given \tilde{B} is:

$$H(P|\tilde{B}) = -\sum_{i=1}^c p(\tilde{A}_i|\tilde{B}) \log p(\tilde{A}_i|\tilde{B}) = -\sum_{i=1}^c p(\tilde{A}_i\tilde{B})/p(\tilde{B}) \log(p(\tilde{A}_i\tilde{B})/p(\tilde{B})) . \tag{1}$$

Let $Q = \{Q_1, Q_2, \dots, Q_n\}$ be a natural fuzzy partition, $P = \{\tilde{A}_1, \tilde{A}_2, \dots, \tilde{A}_c\}$ be a fuzzy partition of $X = \{X_1, X_2, \dots, X_n\} \subset R^p$, the entropy of fuzzy partition P is [4]:

$$H(P) = \sum_{i=1}^c H(Q|\tilde{A}_i) = -\sum_{i=1}^c \sum_{j=1}^n p(Q_j\tilde{A}_i)/p(\tilde{A}_i) \log(p(Q_j\tilde{A}_i)/p(\tilde{A}_i)) . \tag{2}$$

2.2 Multi-dimensional Fuzzy Partition

Introducing partial order \leq in $[0,1]^n$, $X \leq Y$ if and only if $x_i \leq y_i$ ($i = 1, \dots, n$) for arbitrary $X, Y \in [0,1]^n$. Then introduce three transforms $p_i, q_i, \sigma_{ij} : [0,1]^n \rightarrow [0,1]^n$ as:

$$\begin{aligned} p_i(X) &= (1, \dots, 1, x_i, 1, \dots, 1), \quad q_i(X) = (0, \dots, 0, x_i, 0, \dots, 0), \\ \sigma_{ij}(X) &= \sigma_{ij}(x_1, \dots, x_i, \dots, x_j, \dots, x_n) = (x_1, \dots, x_{i-1}, x_j, x_{i+1}, \dots, x_{j-1}, x_i, x_{j+1}, \dots, x_n) . \end{aligned} \tag{3}$$

Let the mapping $T_n : [0,1]^n \rightarrow [0,1]^n$ be an n-dimensional quasitriangular norm [4]. If T_n satisfies the conditions: $T_n(0, \dots, 0) = 0$, $T_n(1, \dots, 1) = 1$, $T_n(\sigma_{ij}(X)) = T_n(X)$ and $X \leq Y \Rightarrow T_n(X) \leq T_n(Y)$, $T_n(T_n(X), x_{n+1}, \dots, x_{2n-1}) = T_n(x_1, \dots, x_{n-1}, T_n(Y))$ for arbitrary $X = \{x_1, x_2, \dots, x_n\}$, $Y = \{x_n, \dots, x_{2n-1}\}$. If the n-dimensional quasitriangular norm T_n satisfies $T_n(p_i(X)) = x_i$, T_n is a n-dimensional triangular norm. The following mapping of $[0,1]^n$ to $[0,1]$ is a n-dimensional triangular norm:

$$\prod : X \mapsto \prod (X) = \prod_{i=1}^n x_i . \tag{4}$$

Definition 1. If $\tilde{A}_1, \tilde{A}_2, \dots, \tilde{A}_n$ are the fuzzy sets on X_1, X_2, \dots, X_n respectively and uncorrelated, $\int_{X_1 \times X_2 \times \dots \times X_n} T_n(\mu_{\tilde{A}_1}(x_1), \mu_{\tilde{A}_2}(x_2), \dots, \mu_{\tilde{A}_n}(x_n)) / (x_1, x_2, \dots, x_n)$, the direct product of $\tilde{A}_1, \tilde{A}_2, \dots, \tilde{A}_n$, noted as $\tilde{A}_1 \times \tilde{A}_2 \times \dots \times \tilde{A}_n$, is a fuzzy set on $X_1 \times X_2 \times \dots \times X_n$, and $\mu_{\tilde{A}_1 \times \tilde{A}_2 \times \dots \times \tilde{A}_n}(x_1, x_2, \dots, x_n) = T_n(\mu_{\tilde{A}_1}(x_1), \mu_{\tilde{A}_2}(x_2), \dots, \mu_{\tilde{A}_n}(x_n))$, where $T_n(\bullet)$ is a n-dimensional triangular norm [4].

Theorem 1. Let $P^i = \{\tilde{A}_1^i, \tilde{A}_2^i, \dots, \tilde{A}_{c_i}^i\}$ be a fuzzy partition on $X^i = \{x_1^i, x_2^i, \dots, x_{m_i}^i\} \subset R$ ($i = 1, 2, \dots, n$) and $\forall i \neq j = 1, 2, \dots, n$, \tilde{A}_i^i and \tilde{A}_j^j are uncorrelated ($t_i = 1, 2, \dots, c_i$; $t_j = 1, 2, \dots, c_j$), then $P^1 \times P^2 \times \dots \times P^n = \{\tilde{A}_1^1 \times \tilde{A}_2^2 \times \dots \times \tilde{A}_{c_i}^i \mid t_i = 1, 2, \dots, c_i, \mid i = 1, 2, \dots, n\}$ is a fuzzy partition on $X^1 \times X^2 \times \dots \times X^n \subset R^n$ under \prod , where the membership function of $\tilde{A}_1^1 \times \tilde{A}_2^2 \times \dots \times \tilde{A}_{c_i}^i$ is $\prod_{i=1}^n \tilde{A}_i^i(x^i)$ [4].

3 The Proposed Algorithm for Three-Level Image Segmentation

Let image have L gray levels $G_x = \{0, 1, \dots, L-1\}$ and the average gray h_k in the $s \times s$ neighborhood also have L gray levels $G_y = \{0, 1, \dots, L-1\}$. The 2-D histogram $h(i, j) = h_{ij}$, where i is the gray level value and j is the average gray value in the $s \times s$ neighborhood, and $0 \leq i, j \leq L-1$. Let $\Omega = G_x \times G_y$, the probability of the pixel (i, j) $p(\{[i, j]^T\}) = h_{ij}$. Let the image be composed of three parts, namely dark part \tilde{D} , gray part \tilde{M} and white part \tilde{B} . Membership function $\mu_d(k; a_1, b_1, c_1)$ of \tilde{D} composed of pixels with low gray levels is a Z-function [3]; membership function $\mu_m(k; a_1, b_1, c_1; a_2, b_2, c_2)$ of \tilde{M} composed of pixels with medium gray levels is a Π -function [3]; membership function $\mu_b(k; a_2, b_2, c_2)$ of \tilde{B} composed of pixels with high gray levels is a S-function [3], with conditions $a_1 \leq b_1 \leq c_1 \leq a_2 \leq b_2 \leq c_2$, where k is the gray level in Q . The parameters $a_1, b_1, c_1, a_2, b_2, c_2$ control the shapes of the membership functions, and satisfy $\mu_d(k; a_1, b_1, c_1) + \mu_m(k; a_1, b_1, c_1; a_2, b_2, c_2) + \mu_b(k; a_2, b_2, c_2) = 1$. $\{\tilde{D}, \tilde{M}, \tilde{B}\}$ is a fuzzy partition of 1-D space G_x , as shown in Fig. 1(a).

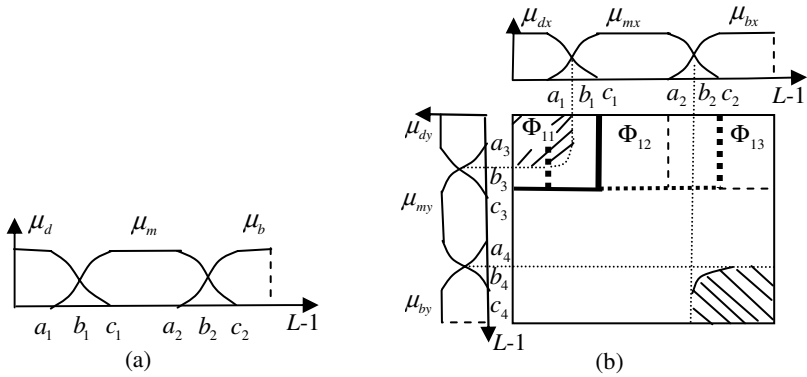


Fig. 1. (a) Fuzzy partition of 1-D; (b) Fuzzy partition of 2-D histogram

Fig. 1(b) shows a fuzzy partition of 2-D space. The membership functions $\mu_{dx}(i; a_1, b_1, c_1)$, $\mu_{mx}(i; a_1, b_1, c_1, a_2, b_2, c_2)$, $\mu_{bx}(i; a_2, b_2, c_2)$ and $\mu_{dy}(j; a_3, b_3, c_3)$, $\mu_{my}(j; a_3, b_3, c_3, a_4, b_4, c_4)$, $\mu_{by}(j; a_4, b_4, c_4)$ with the same form as $\mu_d(k; a_1, b_1, c_1)$, $\mu_m(k; a_1, b_1, c_1, a_2, b_2, c_2)$, $\mu_b(k; a_2, b_2, c_2)$, respectively, are used to get a corresponding fuzzy partition of G_x and G_y , where i and j are independent free variables. The parameters $(a_1, b_1, c_1; a_2, b_2, c_2; a_3, b_3, c_3; a_4, b_4, c_4)$ control the shapes of the six membership function, with conditions $a_1 \leq b_1 \leq c_1 \leq a_2 \leq b_2 \leq c_2$ and $a_3 \leq b_3$

$\leq c_3 \leq a_4 \leq b_4 \leq c_4$. Then nine fuzzy sets of the sample space $\Omega = G_x \times G_y$ are obtained. Each set is the intersection set of the six membership functions of G_x and G_y , as shown in Fig. 1(b): the thick real line region Φ_{11} , the thick dotted line region Φ_{12} , the fine dotted line Φ_{13} , and the other six regions $\Phi_{21}, \Phi_{22}, \Phi_{23}, \Phi_{31}, \Phi_{32}, \Phi_{33}$ obtained in the same way. It is easy to get the proof that $\{\Phi_{11}, \Phi_{12}, \Phi_{13}, \Phi_{21}, \Phi_{22}, \Phi_{23}, \Phi_{31}, \Phi_{32}, \Phi_{33}\}$ is a fuzzy partition of the sample space $\Omega = G_x \times G_y$ based on the Theorem 1. Based on Definition 1, with the triangular norm \prod in formula (4), the membership functions of the nine fuzzy sets are given as:

$$\begin{aligned} \mu_{\Phi_{11}} &= \mu_{dx}(i)\mu_{dy}(j) = \mu_{11}(i, j), \mu_{\Phi_{12}} = \mu_{dx}(i)\mu_{my}(j) = \mu_{12}(i, j), \mu_{\Phi_{13}} = \mu_{dx}(i)\mu_{by}(j) = \mu_{13}(i, j), \\ \mu_{\Phi_{21}} &= \mu_{mx}(i)\mu_{dy}(j) = \mu_{21}(i, j), \mu_{\Phi_{22}} = \mu_{mx}(i)\mu_{my}(j) = \mu_{22}(i, j), \mu_{\Phi_{23}} = \mu_{mx}(i)\mu_{by}(j) = \mu_{23}(i, j), \\ \mu_{\Phi_{31}} &= \mu_{bx}(i)\mu_{dy}(j) = \mu_{31}(i, j), \mu_{\Phi_{32}} = \mu_{bx}(i)\mu_{my}(j) = \mu_{32}(i, j), \mu_{\Phi_{33}} = \mu_{bx}(i)\mu_{by}(j) = \mu_{33}(i, j). \end{aligned} \tag{5}$$

Let $\tilde{D} = \Phi_{11}$, $\tilde{M} = \Phi_{12} \cup \Phi_{21} \cup \Phi_{22}$, $\tilde{B} = \Phi_{13} \cup \Phi_{23} \cup \Phi_{31} \cup \Phi_{32} \cup \Phi_{33}$. The membership functions of the fuzzy sets \tilde{D} , \tilde{M} , \tilde{B} are as following:

$$\begin{aligned} \mu_{\tilde{D}}(i, j) &= \mu_{11}(i, j), \mu_{\tilde{M}}(i, j) = \mu_{12}(i, j) + \mu_{21}(i, j) + \mu_{22}(i, j), \\ \mu_{\tilde{B}}(i, j) &= \mu_{13}(i, j) + \mu_{23}(i, j) + \mu_{31}(i, j) + \mu_{32}(i, j) + \mu_{33}(i, j). \end{aligned} \tag{6}$$

$P = \{\tilde{D}, \tilde{M}, \tilde{B}\}$ is a fuzzy partition of the sample space $\Omega = G_x \times G_y$ based on Theorem 1. Construct a sequence of sets $Q_{i,j} = \{[i, j]^T, i, j = 0, 1, \dots, L-1\}$. $Q = \{Q_{00}, \dots, Q_{(L-1)(L-1)}\}$ is a natural fuzzy partition on $\Omega = G_x \times G_y$. Based on the formula (1), the conditional entropy of the natural fuzzy partition Q given \tilde{D} is:

$$H(Q|\tilde{D}) = -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \frac{p(Q_{i,j}|\tilde{D})}{p(\tilde{D})} \log \frac{p(Q_{i,j}|\tilde{D})}{p(\tilde{D})} = -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \frac{\mu_{\tilde{D}}(i, j)h_{i,j}}{p(\tilde{D})} \log \frac{\mu_{\tilde{D}}(i, j)h_{i,j}}{p(\tilde{D})}. \tag{7}$$

The conditional entropy of the natural fuzzy partition Q given \tilde{M} is:

$$H(Q|\tilde{M}) = -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \frac{p(Q_{i,j}|\tilde{M})}{p(\tilde{M})} \log \frac{p(Q_{i,j}|\tilde{M})}{p(\tilde{M})} = -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \frac{\mu_{\tilde{M}}(i, j)h_{i,j}}{p(\tilde{M})} \log \frac{\mu_{\tilde{M}}(i, j)h_{i,j}}{p(\tilde{M})}. \tag{8}$$

The conditional entropy of the natural fuzzy partition Q given \tilde{B} is:

$$H(Q|\tilde{B}) = -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \frac{p(Q_{i,j}|\tilde{B})}{p(\tilde{B})} \log \frac{p(Q_{i,j}|\tilde{B})}{p(\tilde{B})} = -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \frac{\mu_{\tilde{B}}(i, j)h_{i,j}}{p(\tilde{B})} \log \frac{\mu_{\tilde{B}}(i, j)h_{i,j}}{p(\tilde{B})}. \tag{9}$$

where $p(\tilde{D}) = -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \mu_{\tilde{D}}(i, j)h_{i,j}$, $p(\tilde{M}) = -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \mu_{\tilde{M}}(i, j)h_{i,j}$, $p(\tilde{B}) = -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \mu_{\tilde{B}}(i, j)h_{i,j}$.

Based on (1), (7), (8), (9), the entropy of fuzzy partition P is:

$$\begin{aligned}
 H(P) = H(Q|\tilde{D}) + H(Q|\tilde{M}) + H(Q|\tilde{B}) &= -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \left[\frac{\mu_{\tilde{D}}(i, j)h_{i,j}}{p(\tilde{D})} \log \frac{\mu_{\tilde{D}}(i, j)h_{i,j}}{p(\tilde{D})} \right] \\
 -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \left[\frac{\mu_{\tilde{M}}(i, j)h_{i,j}}{p(\tilde{M})} \log \frac{\mu_{\tilde{M}}(i, j)h_{i,j}}{p(\tilde{M})} \right] &- \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \left[\frac{\mu_{\tilde{B}}(i, j)h_{i,j}}{p(\tilde{B})} \log \frac{\mu_{\tilde{B}}(i, j)h_{i,j}}{p(\tilde{B})} \right]. \tag{10}
 \end{aligned}$$

The regions far away from the diagonal line in the 2-D histogram can be neglected which represent the noise of the image and always have little probability. Therefore, we can get the simplified formula:

$$\begin{aligned}
 H(P) = H(Q|\tilde{D}) + H(Q|\tilde{M}) + H(Q|\tilde{B}) &= -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \left[\frac{\mu_{11}(i, j)h_{i,j}}{p(\tilde{D})} \log \frac{\mu_{11}(i, j)h_{i,j}}{p(\tilde{D})} \right] \\
 -\sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \left[\frac{\mu_{22}(i, j)h_{i,j}}{p(\tilde{M})} \log \frac{\mu_{22}(i, j)h_{i,j}}{p(\tilde{M})} \right] &- \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} \left[\frac{\mu_{33}(i, j)h_{i,j}}{p(\tilde{B})} \log \frac{\mu_{33}(i, j)h_{i,j}}{p(\tilde{B})} \right]. \tag{11}
 \end{aligned}$$

$H(Q|\tilde{D})$, $H(Q|\tilde{M})$, $H(Q|\tilde{B})$ can be regarded as the information measure of the dark, gray and white part of the image, respectively. The formula (11) can be regarded as the overall information measure of the image based on the fuzzy partition $P = \{\tilde{D}, \tilde{M}, \tilde{B}\}$. The algorithm is aimed at searching for maximum information based on a fuzzy partition, namely searching for an optimal combination of 12 parameters $(a_1, b_1, c_1; a_2, b_2, c_2; a_3, b_3, c_3; a_4, b_4, c_4)$ so that the total fuzzy entropy $H(P)$ in formula (11) has the maximum value. Then the pixels can be classified based on the maximum membership principle: if $\mu_{\tilde{D}}(i, j) \geq 0.5$, the pixel (i, j) is classified to the dark part \tilde{D} , the shadow region in top left corner of the 2-D histogram as shown in Fig.1(b); if $\mu_{\tilde{B}}(i, j) \geq 0.5$, the pixel (i, j) is classified to the white part \tilde{B} , the shadow region in lower right corner of the 2-D histogram as shown in Fig. 1(b); otherwise classified to \tilde{M} . Because of the large number of possible combinations of the 12 parameters in a multi-dimensional fuzzy partition, it is not practical to compute each possible value. QGA can be used to find the optimal combination.

4 Parameter Optimization by QGA

In this section, we will briefly state the principle of QGA and explain how to use QGA to find a combination of the 12 membership parameters such that $H(P)$ has the maximum value.

4.1 Coding Method and Quantum Gate of QGA

The basic unit of information in quantum computation is the qubit [5]. A qubit is a two-level quantum system and it can be represented by a unit vector of a two dimensional Hilbert space $(\alpha, \beta \in \mathbb{C}) : |\Psi\rangle = \alpha |0\rangle + \beta |1\rangle$, $|\alpha|^2 + |\beta|^2 = 1$, where we denote the basis states with $|0\rangle$ and $|1\rangle$, adopting the ket notation for quantum state

vectors. $|\alpha|^2$ and $|\beta|^2$ represent the probability of the basic state 0 and 1 of the system respectively. A qubit chromosome p of the length k is defined in formula (13), which can represent 2^k probable states.

$$p = \begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_k \\ \beta_1 & \beta_2 & \cdots & \beta_k \end{bmatrix}. \tag{12}$$

Qubit chromosome is updated by some appropriate quantum gate $U(\theta)$ [5], which is defined as:

$$U(\theta_i) = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \end{bmatrix}, \quad i = 1, 2, \dots, k. \tag{13}$$

The quantum gate is updated by $[\alpha'_i \ \beta'_i]^T = U(\theta_i)[\alpha_i \ \beta_i]^T$, where (α_i, β_i) is the i -th qubit, θ_i is the rotation angle. A general updating strategy is used in this paper, where the rotation angle selection strategy is referred to [5]. The updating strategy is to compare the fitness $g(x_i)$ of the current measured value of the individual x_i with the present evolutionary aim's fitness $g(b_i)$ of the current optimal individual b_i . If $g(x_i) > g(b_i)$, then adjust the qubit of the corresponding bit ($x_i \neq b_i$) to make the probability amplitude evolve toward the direction benefiting the appearance of x_i . On the contrary, if $g(x_i) < g(b_i)$, then adjust the qubit of the corresponding bit to make the probability amplitude evolve toward the direction benefiting the appearance of b_i .

4.2 The Structure of QGA

(1) Initialize colony $Q(s)$: $Q(s) = \{p_1^s, p_2^s, \dots, p_n^s\}$ are initialized to $(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ [5], where n is the size of population and $p_j^s, (j = 1, 2, \dots, n)$ represents the j -th solution in the s -th generation in the colony with the length k of the qubits; (2) Make $P(s)$ by measuring $Q(s)$ states: generate a random number r between 0 and 1. If $r > |\alpha_i|^2$, the measuring result is 1; otherwise 0; (3) Evaluate $P(s)$ and store the best individual among $P(s)$ and its fitness; (4) Update $Q(s)$ using quantum gates, get son colony $Q(s+1)$; $s = s+1$ and turn to (3). The preset evolutionary generation T is used as iteration stopping criterion in this paper.

4.3 Initial Setup of QGA by Experiments

Instead of searching blindly, an effective searching scheme is generated using the necessary condition to make the entropy function $H(P)$ in formula (11) arrive at the maximum value. The first step is to encode the parameters $a_1, b_1, c_1; a_2, b_2, c_2$;

$a_3, b_3, c_3; a_4, b_4, c_4$ into an alphabet string, notice that have to following the increasing order $0 \leq a_1 \leq b_1 \leq c_1 \leq a_2 \leq b_2 \leq c_2 \leq 255$ and $0 \leq a_3 \leq b_3 \leq c_3 \leq a_4 \leq b_4 \leq c_4 \leq 255$, referring to the fuzzy membership functions [3]. Here we use a mathematical processing method [3]:

$$\begin{aligned} c_1^1 &= c_1, b_1^1 = c_1^1 * (b_1 / 255), a_1^1 = b_1^1 * (a_1 / 255), \\ a_2^1 &= c_1^1 + (255 - c_1^1) * (a_2 / 255), b_2^1 = a_2^1 + (255 - a_2^1) * (b_2 / 255), \\ c_2^1 &= b_2^1 + (255 - b_2^1) * (c_2 / 255). \end{aligned} \tag{14}$$

Then the condition $0 \leq a_1^1 \leq b_1^1 \leq c_1^1 \leq a_2^1 \leq b_2^1 \leq c_2^1 \leq 255$ is satisfied. And the other six parameters are processed in the same way to satisfy the condition $0 \leq a_3^1 \leq b_3^1 \leq c_3^1 \leq a_4^1 \leq b_4^1 \leq c_4^1 \leq 255$.

The parameters of QGA are set as following by experiments on a test image: the dimension of the parameters $D=12$, the length of qubit string $k=12 \times 8=96$, the size of population $n=30$, and the maximal generation number $T=70$. We choose the entropy function in formula (11) as the fitness function.

5 Experimental Results

The proposed algorithm has been tested using a gray image and we compared our results with that using Tao's method in Fig. 2. Fig. 2(a) shows the gray scale pills image with size 448x646. Gaussian noise with the mean value 0 and the variance 0.002 is injected into the image. Simulation experiments are worked on PC with Matlab7, Pentium processor 1.6GHz CPU and 1G memory.

Fig. 2(b) shows the three-level segmentation result using Tao's method with time 9.32(s), while the parameters of QGA are set as following: $D=6, k=48, n=30, T=100$. The optimal combination of the membership parameters $(a_1, b_1, c_1; a_2, b_2, c_2) = (0, 2, 128, 132, 145, 248)$, while the thresholds are (38, 171). The correspondent shapes of the membership functions are shown in Fig. 2(d). Fig. 2(c) shows the experiment result with the proposed method by QGA with time 53.12(s). The optimal combination $(a_1, b_1, c_1; a_2, b_2, c_2, a_3, b_3, c_3; a_4, b_4, c_4) = (18, 30, 64, 91, 122, 134, 20, 27, 83, 119, 120, 125)$. Fig. 2(e) shows the corresponding 6 shapes of the membership functions and its fuzzy partition of the 2-D histogram. Comparing the thresholded images of Fig. 2(b), (c), one can observe better visual effect using the proposed method than that using Tao's method for three-level image segmentation. Using a uniformity measurement [9], an image quality evaluation criterion, the value of Tao's method is 0.7315 and the value of our method is 0.7971, which means that our method has relatively better segmentation effect. Theoretically, the image is segmented to three parts, corresponding to three gibbous areas (the shadow regions in Fig. 1(b)), which are enclosed by straight line segments and high order hyperbolic curves. The regions enclosed by high order hyperbolic curves are located in diagonal direction, which are the key regions to distinguish the three parts. Thus, our method can get good segmentation effect.

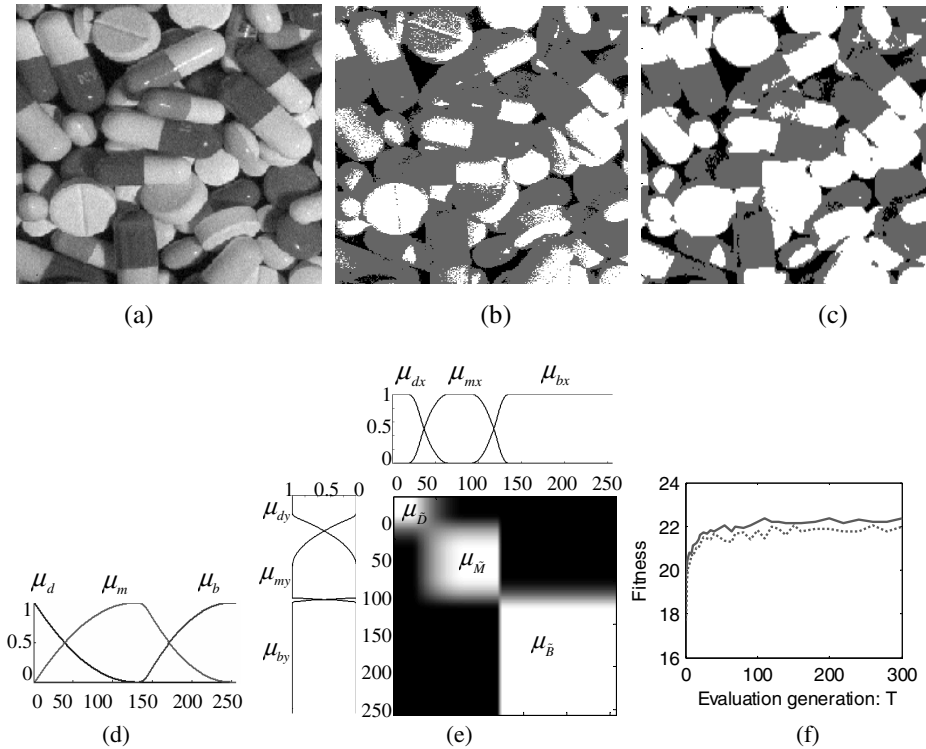


Fig. 2. (a) Gray scale pills image; (b) Result image using Tao’s method; (c) Result image using the proposed method; (d) Member function plot using Tao’s method; (e) Membership function plot and its fuzzy partition of the 2-D histogram using the proposed method; (f) Comparison diagram of the convergence curves of QGA (*real line*) and GA (*dotted line*)

In the same time, in order to evaluate the performances of the QGA, we use GA [6] to search the optimal combinations of the 12 parameters for the same test image. The corresponding parameters D , k , n , T and fitness function of GA are the same as QGA. Other parameters of GA are set as following: the probability of crossover $P_c=0.5$, the probability of mutation $P_m=0.01$. Fig. 2(f) shows the convergence curves of the fuzzy partition entropy with QGA and GA by 10 times average method. The curve of the fitness of QGA displays a fast convergence speed than GA with the same evaluation generations. And the fitness of QGA converges to 22.1780, comparing to 21.9849 of GA, with the maximal generation number $T=1000$. It shows that QGA is more effective to solve the optimization problem than GA.

6 Conclusions

In this paper we present a three-level image segmentation method based on multi-dimensional fuzzy partition and fuzzy entropy theory. We define a new fuzzy entropy based on a new fuzzy partition of 2-D histogram. The image is segmented to three

parts, including dark, gray and white part, which are represented by a group of fuzzy sets $P = \{\tilde{D}, \tilde{M}, \tilde{B}\}$. There are 12 parameters in those member functions, determining the attributes of the fuzzy region of the image. The fuzzy region is found by an optimization method (QGA) based on the maximum fuzzy entropy principle, with appropriate coding method to avoid useless chromosome. The experiment results show that our proposed method can get better image segmentation effect than Tao's method. However, it is time consuming for the search of the best sets of the 12 parameters. So the next job can be done for this problem.

Acknowledgments. This work is supported by National Nature Science Foundation of China (Grant No. 60572133).

References

1. Cheng, H.D., Chen, J., Li, J.: Fuzzy Homogeneity Approach to Multilevel Thresholding. *IEEE Trans. Image Processing* 7(7), 1084–1088 (1998)
2. Zhao, M., Fu, A.M.N., Yan, H.: A technique of three-level thresholding based on probability partition and fuzzy 3-partition. *IEEE Trans. Fuzzy Systems* 9(3), 469–479 (2001)
3. Tao, W.B., Tian, J.W., Liu, J.: Image Segmentation by Three-level Thresholding Based on Maximum Fuzzy Entropy and Genetic Algorithm. *Pattern Recognition Letters* 24(16), 3069–3078 (2003)
4. Jin, L.Z., Yuan, X.H., Zhao, Y.F., Xiao, L.Z.: Image Segmentation Through Maximizing Fuzzy Partition Entropy of 2-D histogram (in Chinese). *Journal of Electronics and Information Technology* 24(8), 1040–1046 (2002)
5. Yang, J., Li, B., Zhuang, Z.Q.: Research of Quantum Genetic Algorithm and Its Application in Blind Source Separation. *Journal of Electronics* 20(1), 62–68 (2003)
6. Bhanu, B., Lee, S., Ming, J.: Adaptive Image Segmentation Using a Genetic Algorithm. *IEEE Trans systems, man, and cybernetics* 25(12), 1543–1567 (1995)
7. Zadeh, L.A.: Probability Measures of Fuzzy Events. *Journal Math. Anal. Appl.* 23(3), 421–427 (1968)
8. Ruspini, E.H.: A New Approach to Clustering. *Inform. Control* 15(1), 22–32 (1969)
9. Sahoo, P.K., Soltani, S., Wong, A.K.C., et al.: A Survey of Thresholding Techniques. *Computer Vision, Graphics and Image processing* 41, 233–260 (1988)

A Hybrid Particle Swarm Optimization for Feed-Forward Neural Network Training

Ben Niu* and Li Li*

School of Management, Shenzhen University,
Shenzhen, 518060 P.R. China
drniuben@gmail.com, llii@163.com

Abstract. This paper employs a hybrid particle swarm optimization using optimal foraging theory (PSOFT) for multilayer feed-forward neural network (MFNN) training. Three benchmark classification problems: Iris, New-thyroid and Glass are conducted to measure the performance of PSOFT based MFNN. The simulation results are also compared with obtained using back Propagation (BP), genetic algorithm (GA) and standard PSO (SPSO) approaches to demonstrate the effectiveness and efficiency of PSOFT.

Keywords: Hybrid particle swarm optimization; neural network; training.

1 Introduction

Artificial neural networks (ANNs), and in particular, multi-layer feed-forward neural networks (multi-layer perceptron networks), have been widely applied in different fields. It was claimed that a MFNN can be trained with non-linear transfers to approximate and accurately generalize virtually any smooth, measurable function whilst taking no prior assumptions concerning the data distribution [1]. The most popular algorithm for training MFNNs is error back propagation (BP). Due to BP algorithm is based on the gradient information of an objective function, it possesses the disadvantages of slowly convergence speed and easy entrapment in a local minima. Currently, the rising of heuristic algorithms such as simulated annealing (SA), genetic algorithm (GA) and evolutionary strategy (ES) has inspired as new resources to train MFNNs.

Particle swarm optimization (PSO) is a relatively new heuristic algorithm developed by Eberhart and Kennedy in 1995 [2, 3]. As PSO possesses the characters of easy implementation, simple realization and rapid convergence capability, it has been utilized widely in many areas, including training neural networks [4]. Although PSO has a fast convergence speed and can find a fairly good solutions at the beginning of search process, it may has great chance to be trapped in a local minima and fails to converge to a global optimum at the end of the search process. To overcome this problem, various attempts have been made to improve the performance of basic PSO, such as tuning the parameters in the velocity and position update

* Corresponding authors.

equations of PSO[5, 6, 7], designing different population topologies [8, 9, 10, 11], combining PSO with other search techniques [12, 13] and incorporating bio-inspired mechanisms into the basic PSO[14, 15].

Nature always presents us with a wide variety of simple biological models which have the bonus effect of increasing our knowledge of how to design more powerful intelligent algorithm. Because the underlying idea of PSO is to mirror the social behavior of a flock of birds during the search of food, it is natural to think that if some biological models in optimal foraging theory can be employed to improve the performance of SPSO. Based on the research of OFT, we introduce two biological-inspired mechanisms in OFT to improve the performance of original PSO, and this proposed method is named PSOOFT[16]. The first mechanism is based on an analogy with reproduction in nature and aims to enhance the ability of convergence rate. The second mechanism is based on a patch choice scheme found in animal groups, where the particles (birds) are attracted to the position of the best particle with a predefined probability and simultaneously migrate to other palace in the searcher domain with another predefined probability. Experiments on five benchmark functions demonstrated PSOOFT is superior to SPSO in both solution quality and convergence rate.

The motivation of this paper is to apply PSOOFT as a new learning algorithm to train a MFNN instead of traditional BP algorithm. PSOOFT will be used to train the MFNN weights (bias) for three benchmark classification problems. For evaluating the performance of the proposed method, GA and BP algorithm will be used to tackle the same problem to illustrate the effectiveness and competitiveness of PSOOFT.

The remainder of the paper is organized as follows. Section 2 presents the standard PSO. The PSOOFT algorithm is introduced in section 3. Section 4 presents the experimental settings and experimental results. Finally, in section 5 we summarize our conclusions and future works.

2 Review of Standard PSO (SPSO)

In PSO, the potential solutions, called particles, fly in a D -dimension search space with a velocity that is dynamically adjusted according to its own experience and that of its neighbors. The i th particle is represented as $\vec{x}_i = (x_{i1}, x_{i2}, \dots, x_{iD})$, where $x_{id} \in [l_d, u_d]$, $d \in [1, D]$, l_d, u_d are the lower and upper bounds for the d th dimension, respectively. The velocity for particle i is represented as $\vec{v}_i = (v_{i1}, v_{i2}, \dots, v_{iD})$, which is clamped to a maximum velocity vector \vec{v}_{\max} . The best previous position of the i th particle is recorded and represented as $P_i = (P_{i1}, P_{i2}, \dots, P_{iD})$, which is also called *pbest*. The index of the best particle among all the particles in the population is represented by the symbol g , and P_g is called *gbest*. At each iteration step t , the particles are manipulated according to the following equations:

$$v_i(t+1) = w \times v_i(t) + R_1 c_1 (P_i - x_i(t)) + R_2 c_2 (P_g - x_i(t)) \quad (1)$$

$$x_{id} = x_{id} + v_{id}, \quad (2)$$

where w is inertia weight; c_1 and c_2 are acceleration constants; and R_1, R_2 are random vectors with components uniformly distributed in $[0, 1]$.

3 A Novel PSO Based on Optimal Foraging Theory

Foraging theory [17] is based on the assumption that animals search for and obtain nutrients in a way that maximizes their energy intake E per unit time T spent foraging. Hence they try to maximize a function like E/T . Inspired the research of foraging theory, we found that two aspects of OFT is suitable to be incorporated into the SPSO model.

A. Reproduction

The birds incorporate some kind of pressure towards successful behavior, that is, birds that reach valleys must have some kind of reproductive reward, by generating more offspring-or by simply having a higher probability of generating offspring in each time step (suppose that the swarm is requested to find the lower values of one complex function).

After a given time step NR , a reproduction step is taken. Suppose that the population size is S . Let $SR = S/2$ be the number of population members who have had sufficient nutrients so that they will reproduce.

This reproduction procedure is performed as follow. The birds in the population are sorted in ascending order and divided into healthy part and unhealthy part, where the SR birds with relatively lower fitness value are regarded as healthy one and the rest SR birds with higher fitness value the are unhealthy; Then the SR unhealthy birds die and the other SR healthiest birds each split into two young birds. It should be stressed that the personal best information associated with each of the birds is remained unchanged.

B. Patch Choice

To better understand how the patch choice model is incorporated in PSO, consider the following scenario: a group of birds are randomly searching food in an area. There are some pieces of food in the area being searched and only one piece of food is with more nutrients.

Suppose that X is the position of a bird and $F(X) X \in R^D$ represents how much nutrients substances it get. Hence $pbest$ can be regarded as the previous position of the pieces of food searched by the birds and $gbest$ is the position where the best nutrients patch has been found. In each iteration time, the top SR healthy birds among the swarms search for the nearest resource within neighborhood of their previous best positions with probability PC , and migrate to other palace in the searcher domain with probability $1-PC$.

Because the underlying concept of chaotic local search [18] is similar to the bird's local search scheme, it was adopted to mimic the implement of the bird's local search around the neighborhood of the patches.

The overall procedure of PSOOFT is shown in Table 1.

Table 1. The pseudo code for PSOOFT

Initialize step: randomly initialize the position and velocity of each particle
 Set $k := 0$
 While (the termination conditions are not met)
FOR (each particle i in the swarm)
 Calculate fitness: calculate the fitness value of current particle: $F(X_i)$
 Update $pbest$ and $gbest$: compare the fitness value of $pbest$ with $F(X_i)$. If $F(X_i)$ is better than $pbest$, then set the $pbest$ to the current position X_i ; Further more, if $F(X_i)$ is better that $gbest$, the reset $gbest$ to the current index in particle array
 Update the velocity and positions: calculate velocities V_i and position X_i using Eq.1.and Eq.2, respectively
 Limit the velocity: If $V_i > V_{\max}$ then $V_i = V_{\max}$. If $V_i < V_{\min}$ then $V_i = V_{\min}$
 Reproduce particles: If $Mod(k, Nr) == 0$, sort particles fitness value in order. The SR particle with higher values die and the other SR particle with the best value split (and the copies that are made are placed at the same location as their parent).
 Select Patch: If $rand < PC$, chaotic local search is performed, Otherwise migrate to other palace in the searcher domain
END FOR
 Set $k := k + 1$
END WHILE

4 Experimental Studies

In order to demonstrate the performance of the PSOOFT, it is applied to the training of multilayer feed-forward neural networks (MFNNs) for classification problems. The performance of the PSOOFT is also compared with that obtained using BP, GA and SPSO training.

4.1 Training MFNNs Using PSOOFT Algorithm

Upon adopting PSOOFT to train a MFNN, two key problems must be resolved, namely encoding the MFNN and designing the fitness function. For a three-layer MFNN, the free parameters to be coded include weights and biases, which can be defined as a one-dimensional matrix, i.e.,

$$\left\{ \overbrace{1 \ 2 \ \dots \ I \times H}^{w^{(IH)}}, \overbrace{1 \ 2 \ \dots \ H \times O}^{w^{(HO)}}, \overbrace{1 \ 2 \ \dots \ H}^{b^{(H)}}, \overbrace{1 \ 2 \ \dots \ O}^{b^{(O)}} \right\},$$

where I , H and O is the number of neurons in input layer, hidden layer and output layer respectively. $w^{(IH)}$ is a vector of the weights between input layer and hidden layer. $w^{(HO)}$ is a vector of the weights between hidden layer and output layer. $b^{(H)}$ represents a vector of the biases of the hidden layer. $b^{(O)}$ represents a vector of the biases of the output layer. The size of the matrix can be represented by $D=I \times H+H \times O+H+O$.

In particular, each particle in PSOFT contains a set of weights and biases of MFNN. The dimension of each particle is same as the size of the above matrix, i.e. D . The MFNN is trained using PSOFT by moving the particles among the weight space to minimize the mean-squared error (MSE):

$$MSE = \frac{1}{\# patterns} \cdot \frac{1}{O} \sum_{p=1}^{\# patterns} \sum_{K=1}^O (d_{kp} - y_{kp})^2, \tag{3}$$

where d_{kp} is the k -th node of desired output and y_{kp} is the k -th network output.

4.2 Numerical Examples

Three benchmark classification problems, i.e., Iris, New-thyroid and Glass are used for testing. The data sets for those three problems can be obtained from the UCI repository. The network configurations are listed in Table 2.

Table 2. Network configuration

Problem	Architecture	#weights
Iris	4-3-3	27
New-thyroid	5-4-3	39
Glass	9-8-7	143

The SPSO and PSOFT parameters were set to the values $c_1 = c_2 = 2$, and a linearly inertia weight starting at 0.9 and ending at 0.4 was used. The maximum velocity of each particle was set to be half the length of the search space in one dimension. In addition, for PSOFT the parameter NR, PC, K are set as 20, 0.8, 1000, respectively. In GA, the population size is set as 40, the parameters of crossover probability $P_c=0.4$ and the mutation probability $P_m=0.1$ is used. In BP, the learning rate η and the momentum α are set as 0.3 and 0.9, respectively. In each

experiment, each data set was randomly divided into two parts: 2/3 as training set and 1/3 as test set. All results reported below are the averages computed over 10 runs.

Table 3 shows the experimental results averaged 10 runs of the four algorithms. Where Train Correct and Test Correct are the correct rate of classification and generalization averaged 10 runs for the training and test sets, respectively. MSE_t and MSE_g refer to mean square error averaged 10 runs on the training and test set, respectively. It should be note that the BP-based MFNN is evolved for 3000 generations repeated for 10 runs.

Among the four types of methods, PSOOF T achieves the highest classified accuracy in the test part, which demonstrates that the results found by PSOOF T are more stable than that of other methods. The averaged best-so-far MSE (training or test)value over 10 runs generated by PSOOF T are smaller than those generated by any other approaches, which clearly demonstrates the competitiveness of PSOOF T with BP, GA and SPSO.

Table 3. Performance comparisons using different training algorithms

Data set	Algorithm	Train Correct	Test Correct	MSE_t	MSE_g
Iris	BP	0.9550	0.9224	0.0516	0.0591
	GA	0.9332	0.9132	0.0138	0.0186
	PSO	0.9712	0.9628	0.0103	0.0274
	PSOOF T	0.9926	0.9731	0.0099	0.0154
New-thyroid	BP	0.9148	0.7214	0.1823	0.1954
	GA	0.9135	0.8356	0.0144	0.0169
	PSO	0.9642	0.9421	0.0098	0.0252
	PSOOF T	0.9831	0.9568	0.0079	0.0166
Glass	BP	0.6896	0.6142	0.1585	0.1185
	GA	0.7356	0.6241	0.1526	0.1492
	PSO	0.8086	0.6857	0.1248	0.1425
	PSOOF T	0.8623	0.7136	0.0542	0.0716

5 Conclusions

In this paper, a hybrid particle swarm optimization PSOOF T has been used to train the weights and biases of a multi-layer feed-forward neural network. Application examples on three benchmark classification problems are conducted to measure the performance of the proposed method. The simulation results have been compared with those obtained by BP, GA and SPSO training, which demonstrated the effectiveness of the PSOOF T in terms of accuracy and robustness of the results. However, this is only the first step in applying PSOOF T for neural network training.

In our future work, PSOOF T algorithm will be used to train other neural networks including radial basis function networks, recurrent neural networks and fuzzy neural networks.

Acknowledgements

This work is supported by the National Natural Science Foundation of China under Grant No. 70271014.

References

1. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning Internal Representation by Error Propagation. *Parallel Distributed Processing* 1, 318–362 (1986)
2. Eberchart, R.C., Kennedy, J.: A New Optimizer Using Particle Swarm Theory. In: 6th IEEE international symposium on Micromachine and Human Science, pp. 39–43. IEEE Press, Piscataway (1995)
3. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: *Proceeding of IEEE International Conference on Neural Networks*, pp. 1942–1948. IEEE Press, Piscataway (1995)
4. Juang, C.F.: A Hybrid of Genetic Algorithm and Particle Swarm Optimization for Recurrent Network Design. *IEEE Trans. Syst., Man, and Cyber., Part B: Cybernetics* 34, 997–1006 (2004)
5. Shi, Y., Eberhart, R.C.: A Modified Particle Swarm Optimizer. In: *Proceedings of IEEE International Conference on Evolutionary Computation*, pp. 69–73. IEEE Press, Piscataway (1998)
6. Chatterjee, A., Siarry, P.: Nonlinear Inertia Weight Variation for Dynamic Adaptation in Particle Swarm Optimization. *Computers & Operations Research* 33, 859–871 (2006)
7. Clerc, M., Kennedy, J.: The Particle Swarm: Explosion, Stability, and Convergence in a Multidimensional Complex Space. *IEEE Trans. on Evolutionary Computation* 6, 58–73 (2002)
8. Kennedy, J.: Small Worlds and Mega-minds: Effects of Neighborhood Topology on Particle Swarm Performance. In: *Proceedings of the Congress on Evolutionary Computation*, pp. 1931–1938. IEEE Press, Piscataway (1999)
9. Kennedy, J., Mendes, R.: Population Structure and Particle Swarm Performance. In: *Proceedings of the 2002 Congress on Evolutionary Computation*, pp. 1671–1675. IEEE Press, Piscataway (2002)
10. Suganthan, P.N.: Particle Swarm Optimizer with Neighborhood Operator. In: *Proceedings of the Congress on Evolutionary Computation (CEC 1999)*, pp. 1958–1962. IEEE Press, Piscataway (1999)
11. Hu, X., Eberhart, R.C.: Multiobjective Optimization Using Dynamic Neighborhood Particle Swarm Optimization. In: *Proceedings of the Congress on Evolutionary Computation*, pp. 1677–1681. IEEE press, Piscataway (2002)
12. Zhang, W.J., Xie, X.F.: DEPSO: Hybrid Particle Swarm with Differential Evolution Operator. In: *Proc. of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 3816–3821. IEEE Press, Washington (2003)
13. Shi, X.H., Liang, Y.C., Lee, H.P., Lu, C., Wang, L.M.: An Improved GA and a Novel PSO-GA-based Hybrid Algorithm. *Information Processing Letters* 93, 255–261 (2005)
14. He, S., Wu, Q.H., Wen, J.Y., Saunders, J.R., Paton, R.C.: A Particle Swarm Optimizer with Passive Congregation. *Biosystems* 78, 135–147 (2004)
15. Niu, B., Zhu, Y.L., He, X.X., Wu, H.: MCPSO: A Multi-Swarm Cooperative Particle Swarm Optimizer. *Applied Mathematics and Computation* 185, 1050–1062 (2007)

16. Niu, B., Zhu, Y.L., Hu, K.Y., Li, S.F., He, X.X.: A Novel Particle Swarm Optimizer Using Optimal Foraging Theory. In: Huang, D.-S., Li, K., Irwin, G.W. (eds.) ICIC 2006. LNCS (LNBI), vol. 4115, pp. 61–71. Springer, Heidelberg (2006)
17. Stephens, D.W., Krebs, J.R.: Foraging Theory. Princeton University Press, Princeton (1986)
18. Choi, C., Lee, J.: Chaotic Local Search Algorithm. *Artificial Life and Robotics* 2, 41–47 (1998)

A Modified Ant Colony Algorithm Used for Multi-robot Odor Source Localization

Yuhua Zou and Dehan Luo

Faculty of Information Engineering,
Guangdong University of Technology, Guangzhou 510006, P.R. China
dehanluo@gdut.edu.cn

Abstract. A multi-robot cooperation strategy based on a modified ant colony algorithm (ACA) is proposed. It enables the multi-robot system to search for the odor sources, which exist in the indoor environment, by imitating the forage behavior of the ant society. The modification of ACA includes new definitions of pheromone and heuristic function. And two extra search modes, local traversal search and global random search are added to improve the search performance of the robot system. A verification procedure is introduced into the iteration process to localize multiple odor sources. Simulation results have showed that the modified algorithm can effectively enable the robots to approach and determine the odor sources quickly and accurately.

Keywords: Ant colony algorithm; Multi-robot cooperation; Odor source localization.

1 Introduction

Recent years there is an increasing demand to inspect the leakages of hazardous chemicals in many fields, for examples, environmental engineering, survivor rescue and dangerous goods detection. As a result, the problem of odor tracking and odor source localization via using mobile robot system has become the research hotspot.

Typical strategies and methods designed for odor tracking and localization include imitation of animal behavior [1,2], logical determination [3], plume tracking [4], artificial neural network [5], probability estimation [6] and multiple robots cooperation [7,8]. A.T. Hayes et al. [7] presented an investigation of odor localization by groups of autonomous mobile robots using principles of Swarm Intelligence (SI), a computational and behavioral metaphor for solving distributed problems that takes its inspiration from biological examples provided by social insects. R.A. Russell et al. [9] developed a robot system for odor tracking that mimics the ant colony behavior by detecting the remnant gas trail on the ground.

The ant colony algorithm (ACA), which was proposed by Dorigo in early 1990s [10,11], is a rapidly developing bionic optimization algorithm that imitates the SI behavior of the ant society. Until now, ACA has been successfully used in solving many problems, such as path planning, task assignment, production schedule, function optimization and multi-robot cooperation strategy [14]. Ding et al. [12] proposed

a multi-robot cooperation algorithm based on ACA by introducing a key concept called “*stigmergy*” of the ant society. The algorithm can organize different numbers of robots to cooperate on a task according to the task difficulty. Meng et al. [13] presented an improved ACA that combined with genetic algorithm to solve the mobile robot odor localization problem. Simulation results show that the robots can asymptotically approach and finally determine the odor source.

The odor source localization is essentially a target searching problem in odor space. The exact position of the odor source is unknown at the beginning for all robots. Hence it’s important to design and select proper strategies for robots to cooperate, search and determine the target. Inspired by the ants’ forage behavior, a multi-robot cooperation strategy based on modified ACA is proposed here. The details about the modified ACA are presented in section 2. Then section 3 gives the simulation results. At the end are the conclusions and a brief outlook for the research.

2 The Modified Ant Colony Algorithm

The modified ACA includes three stages, which are local traversal search, global search and pheromone update. Additionally, an odor source verification procedure runs per certain intervals, so that the robots can search and determine multiple odor sources quickly. Here, we define the robot which detected the maximal gas concentration value at the current iteration as *the best robot*, and accordingly *the worst robot* detected the minimal concentration value. Note that *the best robot* and *the worst robot* correspond to different robot individuals according to the robots’ detected gas concentration values during the iteration cycle.

2.1 Local Traversal Search

The local area is a circle with a radius of d . Fig.1 shows the defined local area and traversal path. Each robot starts from its current position and visits the four orientation points showed in Fig.1. The robot detects and logs the gas concentration at every position it stays. After all the five positions (including the original position) were visited, the robot compares the logged gas concentration values and returns to the location with the maximal concentration. Then the local search is finished.

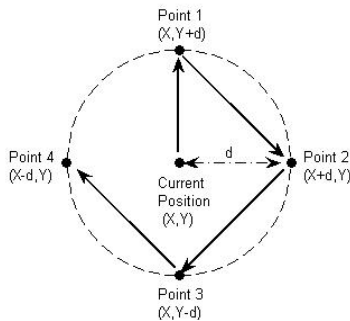


Fig. 1. Local area and traversal sequence of local search

2.2 Global Search

There are two search modes in the global search stage, random search and probability search. After all the robots have finished local search, they communicate with each other about the coordinate and concentration information. Then each robot selects one of the two global modes based on the received information.

Globally Random Search. In order to find the odor source quickly and prevent the system from getting into local optimum, the robot i will move to a stochastic location in the defined environment if the following conditions are satisfied:

$$C_{\max}(t) < C_{\text{thr}} \quad \text{or} \quad C_i(t) = C_{\min}(t) \quad (i = 1, 2, \dots, m) \quad (1)$$

where C_{thr} represents the odor concentration threshold; $C_i(t)$ is the concentration of the location where robot i stays; $C_{\max}(t)$ and $C_{\min}(t)$ are the maximal and minimal concentration of the m locations, respectively. If $C_{\max}(t) < C_{\text{thr}}$ is true, it means that there is still no robot gets close enough to the odor source area and all the robots should go on random searching. If $C_i(t) = C_{\min}(t)$ is satisfied, the robot i will be considered as *the worst robot*, and it goes randomly in order to check that if there is a higher concentration area, preventing the robots to be stagnant in a local optimal zone.

Globally Probability Search. If the robot i does not satisfy the condition (1), it will do global search based on the following probability:

$$p_{ik}(t) = \frac{\tau_k(t)^\alpha \eta_{ik}(t)^\beta}{\sum_{k=1}^m \tau_k(t)^\alpha \eta_{ik}(t)^\beta} \quad (2)$$

where $\tau_k(t)$ denotes the pheromone of the robot k . In our simulation, pheromone is not distributed on the path; instead it is a value indicating each robot's attraction. $\eta_{ik}(t)$ is the heuristic of the robot k to the robot i . α and β are parameters that determine the weightiness of $\tau_k(t)$ and $\eta_{ik}(t)$. The heuristic $\eta_{ik}(t)$ is expressed as:

$$\eta_{ik}(t) = \begin{cases} e^{C_{ik}(t) \cdot D_{ik}(t)} & k \neq i \\ e^{C_i(t)} & k = i \end{cases} \quad (3)$$

where $C_{ik}(t) = C_k(t) - C_i(t)$; $D_{ik}(t)$ is the distance between the robot i and k .

The robot k to which the robot i moves closer is based on the maximal probability $p_{ik}(t)$. The step length is defined as follows:

$$\text{step} = R \cdot D_{ik}(t) \quad (4)$$

where R is a random number with range from 0 to 1.

2.3 Pheromone Updating Rule

When all the robots have finished global search, each robot will do pheromone update. If n ($n < m$) robots move towards the robot i in the global probability search stage and their concentration is improved, the pheromone of the robot i will be updated as follows:

$$\tau_i(t+1) = (1-\rho) \cdot \tau_i(t) + \sum_{k=1}^n C_k(t) \quad (5)$$

where ρ represents the pheromone evaporation factor with range from 0 to 1.

2.4 Odor Source Verification Procedure

The maximal iteration is defined as I_{\max} . When the detected highest concentration value exceeds the threshold C_{thr} for the first time, the odor source verification procedure is activated and will be implemented for every T iterations (we define T as verification cycle). If the locations of the maximal concentrations during the verification cycle T focus on a limited area, and the average value of the maximal concentrations is larger than the verification threshold θ_{avr} , then a new odor source is found and its coordinate equals to the average value of the locations of the maximal concentrations.

After the localization of a new odor source is done, all the robots will be distributed randomly in the defined environment and continue to search for the next odor source. The localized odor source is tagged so that it will not be verified again. The simulation process will be terminated after the I_{\max} iteration cycle is finished.

3 Simulation Results and Analysis

To simplify simulation, the following assumptions are made:

- The robot is thought as a point without mass and size;
- The robot can determine its own position;
- The robot can exactly know (measure) the odor concentration after it arrives at a position;
- Every mobile robot can communicate with other robots;
- The gas/odor sensors' response and recovery delays and communication delays are neglected.

We use Matlab 7.1 software to do the simulation research. A 100 by 100 pixels 2D plan is created to simulate a 10m by 10m indoor environment with a 1m by 1m entrance area at the top left corner. The initial location of each robot is distributed randomly in the entrance area. To facilitate the simulation research, the gas concentration value of each location within the environment is expressed by the following exponential model:

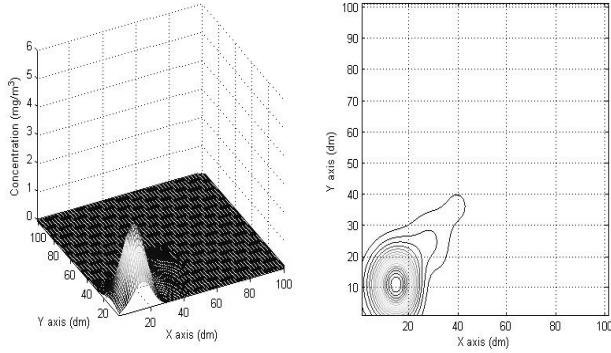


Fig. 2. 3D and 2D concentration maps for a single odor source

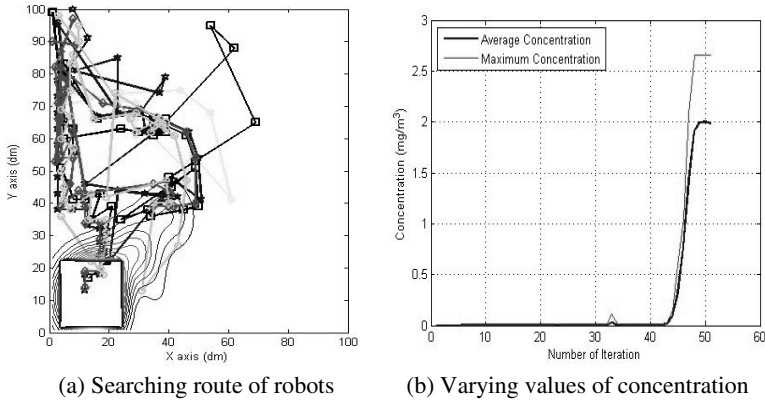


Fig. 3. Searching routes of robots and varying values of concentration without specifying C_{thr}

$$C(x, y) = Q(1 - x)^2 \exp[-x^2 - (y + 1)^2] \tag{6}$$

where $C(x,y)$ represents the concentration value of the location (x,y) ; Q is a parameter that determines the source strength. In our simulation, $m = 5$, $d = 0.5$, $C_{thr} = 0.5$, $\alpha = 1$, $\beta = 10$, $\rho = 0.2$, $I_{max} = 50$, $\theta_{avr} = 1$.

Firstly, we set a single odor source in the simulation environment to validate the importance of C_{thr} and local search. Fig.2 shows the concentration map of the odor source corresponding to the above exponential model with $Q = 1.5$. The strongest concentration point, i.e. the location of the odor source, is located at the point (11,15).

Fig.3(a) shows that without specifying C_{thr} , the robots just swarm together and move laggardly during the whole iteration time. The reason why such phenomena happen is that without specifying C_{thr} , only *the worst robot* can move randomly with a long distance, and the another robots just move step by step with a short step length.

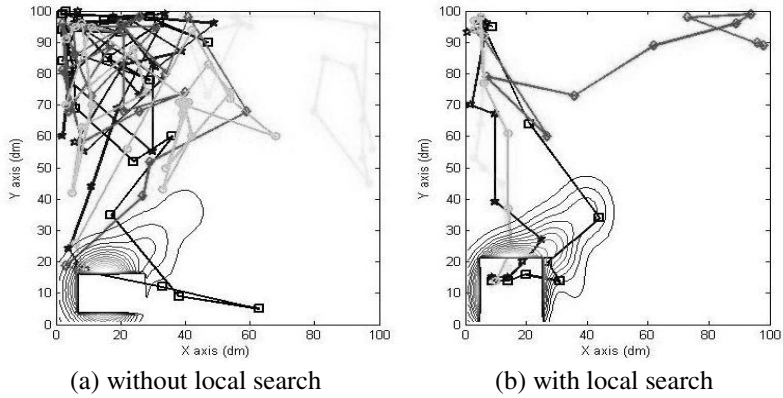


Fig. 4. Comparison of searching routes of robots with different local search strategy

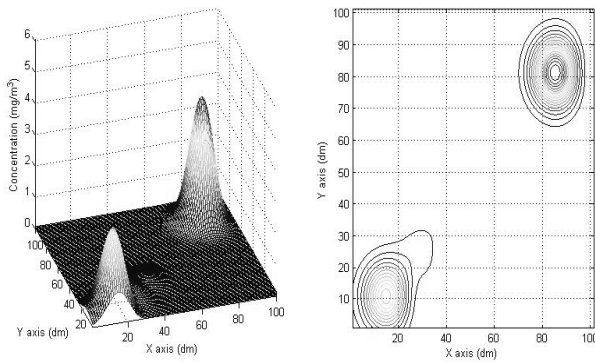


Fig. 5. 3D and 2D concentration maps for two odor sources

Further more, all robots set out collectively from the limited entrance, if they can't move randomly at the beginning, the robots have to go along with each other and it's hard to avoid intervolving. Fig.3(b) shows that it takes a long time for the robots to arrive at the source point without specifying C_{thr} .

Fig.4 compares the searching routes of robots between two search modes. Fig.4(a) shows that without local search stage, the robots move disorderedly at the starting area and take a long time to locate the odor source. By contrast, the routes of robots with local search stage showed in Fig.4(b) are much more ordered, and obviously the robots can find the odor source in shorter time. In addition, it's interesting to see that a robot in Fig.4(b) do not move toward the odor source and keep away from another robots. This robot should be *the worst robot*, and it just moves randomly in the defined environment to find whether there is a new odor source somewhere.

Finally, in order to test the performance of the robot system for multiple odor sources localization, we add a new odor source corresponding to the exponential model (6) with $Q = 2.2$ to the simulation environment. The new source is located at the point (85,81). Fig.5 shows the concentration maps of the two odor sources.

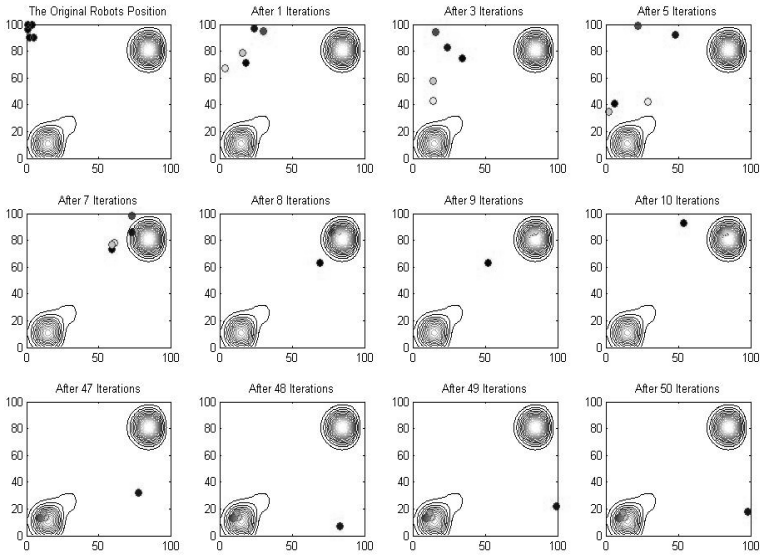


Fig. 6. Distribution maps of robots while searching two sources

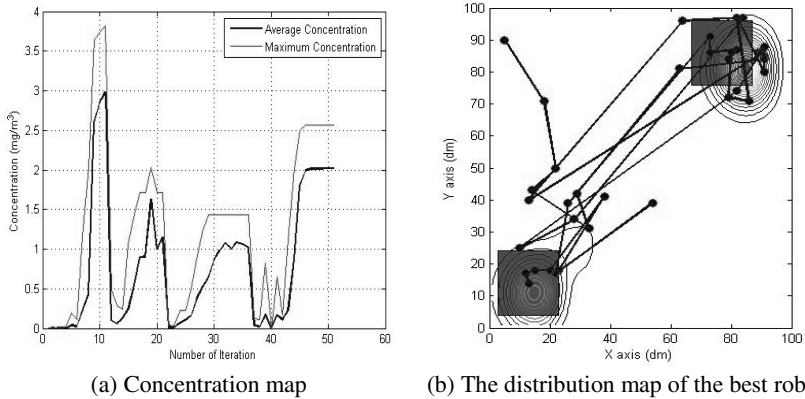


Fig. 7. Concentration map and the distribution map of the best robot

Fig.6 gives the robots’ distribution maps from the underway stage to the odor source verification stage. The robots start moving from the entrance area and gradually disperse. The robots succeed to find the first odor source after 10 iterations and the second one after 50 iterations. Similarly, during the verification stage, there always have a robot, i.e. *the worst robot*, keep searching randomly.

We can find from Fig.7 that during the time between the first and the second odor source verification, the locations of *the best robot* converge near the first verified source for two times. The reason is that the concentration strength of the first source is higher than the second one. So after the first verification, the robots would still move towards the verified source with higher probability. Note that the random distribution of the robots after source verification could affect the result greatly. If a robot

move to the adjacent area of the second source luckily after the random distribution, the verification of the second source will be completed soon.

4 Conclusions

In this paper, a multi-robot cooperation method for odor source localization based on a modified ant colony algorithm is proposed. Simulation results show that the local traversal search and the globally random search can effectively accelerate the searching speed. The modified algorithm inspired by the swarm intelligence of ants' society can successfully find the odor sources exist in the defined environment by using only gas/odor sensors. In the future work, we will try to use and design different concentration models, local search methods and source verification rules to optimize the performance of the robot system.

References

1. Vickers, N.J.: Mechanisms of Animal Navigation in Odor Plumes. *Biological Bulletin* 198, 203–212 (2000)
2. Frank, W.G., Thomas, R.C., David, C.M., Jelle, A.: Biomimetic Robot Lobster Performs Chemo-orientation in Turbulence Using a Pair of Spatially Separated Sensors: Progress and Challenges. *Robotics and Autonomous Systems* 30, 115–131 (2000)
3. Russell, R.A., Thiel, D., Deveza, R., Mackay, S.: A Robotic System to Locate Hazardous Chemical Leaks. In: 1995 IEEE International Conference on Robotics and Automation, pp. 556–561. IEEE Press, New York (1995)
4. Ishida, H., Nakamoto, T., Moriiizumi, T.: Remote Sensing of Gas/Odor Source Location and Concentration Distribution Using Mobile System. *Sen. Acta. B.* 49, 52–57 (1998)
5. Farah, A.M., Duckett, T.: Reactive Localization of an Odor Source by a Learning Mobile Robot. In: 2nd Swedish Workshop on Autonomous Robotics, pp. 29–38 (2002)
6. Farrell, J.A., Murlis, J., Li, W., Carde, R.T.: Filament-based Atmospheric Dispersion Model to Achieve Short Time-Scale Structure of Odor Plumes. *Environmental Fluid Mechanics* 2, 143–169 (2002)
7. Hayes, A.T.: Self Organized Robotic System Design and Autonomous Odor Localization. Ph.D. Dissertation, Computation and Neural Systems, Caltech (2002)
8. Li, J.C., Meng, Q.H., Liang, Q.: Simulation Study on Robot Active Olfaction Based on Evolutionary Gradient Search (in Chinese). *ROBOT* 29, 234–238 (2007)
9. Russell, R.A.: Ant Trails-An Example for Robots to Follow? In: Proc. IEEE International Conference on Robotics and Automation, pp. 2698–2703 (1999)
10. Dorigo, M., Maniezzo, V., Colomi, A.: The Ant System: An Autocatalytic Optimizing Process. Technical Report No. 91-016 Revised, Politecnico di Milano, Italy (1991)
11. Dorigo, M., Maniezzo, V., Colomi, A.: The Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Trans. Syst. Man. Cybern. B.* 26, 29–41 (1996)
12. Ding, Y.Y., He, Y., Jiang, J.P.: Multi-robot Cooperation Method Based on The Ant Algorithm. In: 2003 IEEE Swarm Intelligence Symposium, pp. 14–18. IEEE Press, New York (2003)
13. Meng, Q.H., Li, J.C., Li, F., Zeng, M.: Mobile Robots Odor Localization with an Improved Ant Colony Algorithm. In: 2006 IEEE International Conference on Robotics and Biomimetics, pp. 959–964. IEEE Press, New York (2006)
14. Duan, H.B.: Ant Colony Algorithm: Theory and Applications (in Chinese). Science Press, Beijing (2005)

A New Problem-Solving Method for Multi-agent Systems and Computer Networks: Solitary Wave Propagation

Dianxun Shuai

East China University of Science and Technology, Shanghai 200237, China
shdx411022@online.sh.cn

Abstract. A new problem-solving method for multi-agent systems and computer networks is proposed, which transforms the problem-solving process into the parallel solitary wave propagation process in non-linear space¹. This paper discusses the basic principle of the solitary wave method and proves its properties theoretically. Due to the paper length limitation, the algorithm based on solitary waves and simulations are given in other related papers. The theoretical analysis and practical experiments show the advantages of the proposed method in terms of the high-degree parallelism, the suitability for complex environment, and the powerful processing ability.

Keywords: solitary wave; multi-agent systems; task allocation and resource assignment; bandwidth allocation; parallel optimization.

1 Introduction

The optimization of the network resource allocation is of great importance for enhancing the network throughput and improving the network performance. Since the ATM network is to support multiple classes of traffic (e.g. video, audio, and data) with widely different characteristics and Quality of Service (QoS) requirements, one of the major challenges is to guarantee the promised QoS for all the admitted users, while maximizing the resource utilization through dynamically allocating appropriate resources (e.g. bandwidth, buffers). The bandwidth allocation problem in networks is NP-complete [1]. Many algorithms and evaluation criteria for the bandwidth allocation have been proposed so far [2]-[11], which are based on quite different strategies, such as maximizing an aggregate utility function of users, minimizing the overhead of data transmission, maximizing the average priority or average reliability of traffic, and so forth. Some very interesting duality models are proposed in [12]-[14], by which the network's optimization problem may be solved by a primal parallel algorithm for allotted rates, and a dual parallel algorithm for shadow prices or congestion signals. F. Kelly et al model the problem of rate control for the Internet as a problem of maximizing

¹ This work was supported by the National Natural Science Foundation of China under Grant No.60575040 and No.60473044.

the aggregate utility of users, which could be further decomposed into separate subproblems for the network and for the individual users, resulting in a duality algorithm [12]-[14] that can be carried out in parallel.

As for the optimization problem about the task allocation and resource assignment in multi-agent systems (MAS), it is necessary to determine the cooperative members previously and to form the coalition among many agents before executing the given tasks. The symbolic logic methods currently used in MAS are basically serial or small-scale parallel, with very high computational complexity. Generally, all the agents in MAS are regarded either to be selfish since each of them only pursues its own maximal utility, ignoring the global benefit of the system; or to be unselfish due to their concern only about the whole outcome of the system, without regard for personal payoff. But these are just two extreme cases. Moreover just simple social behaviors, such as cooperation and competition, bilateral and conscious interactions, are considered by most methods currently used in MAS [16]-[18].

This paper is devoted to a new approach to problem-solving in MAS and CN, which is based on the solitary wave propagation (SWP) process in non-linear space. The proposed approach has many advantages over other methods in terms of the high-degree parallelism and the suitability for complex environment. This paper focuses on the theoretical analysis of the solitary wave method. The algorithms and simulations for the parallel optimization are moved in other related papers due to the paper length limitation.

2 Basic Principle of Solitary Wave Mechanism

The solitary wave has both particle and wave properties, and is a universal phenomenon in nature and physics. Particularly, its energy is concentrated in a relatively smaller region, and its waveform and/or wave speed could recover, called as elastic dissemination, when the waves interact mutually. For the parallel optimization of task allocation and resource assignment in MAS and CN, this paper proposes a special kind of solitary wave which propagates concurrently in nonlinear media as follows. The propagation paths and speeds all depend on the competition results between waves. Only the competition winner wave along a hyper-edge can continually propagate further, whereas the loser wave along a hyper-edge is deprived of propagating forwards unless the wave along the hyper-edge becomes a winner again in the competition turn that follows. There are no interference between waves confluent to the same wave node and no reflection from either wave node or hyper-edge. The waveform remains rectangular without any distortion due to dispersion or diffusion. The wave amplitude decreases in inverse proportion to the propagation distance. The wave speed can change with the received wave amplitude and introduced heuristic knowledge.

For a bottom-up implicit AND/OR graph $G(N, E)$, the set of all the input hyper-edges of node $n \in N$ is denoted by $\Omega(n) = \{\mathcal{L}_{n_1}, \dots, \mathcal{L}_{n_k}\}$, where the i -th input hyper-edge $\mathcal{L}_{n_i} = \{\langle m_{n_i}^1, n \rangle, \dots, \langle m_{n_i}^p, n \rangle\}$, and $\langle m_{n_i}^1, n \rangle, \dots, \langle m_{n_i}^p, n \rangle \in E$, $i \in \{1, \dots, k\}$; The set of nodes, $\mathcal{N}_i(n) = \{m_{n_i}^1, \dots, m_{n_i}^p\}$, is called the i -th input hyper-node of node n . Thus the input hyper-edge \mathcal{L}_{n_i} is also described

as the edge from the input hyper-node $\mathcal{N}_i(n)$ to node n , i.e. $\mathcal{L}_{n_i} = \langle \mathcal{N}_i(n), n \rangle$. In the graph $G(N, E)$, a wave source node that represents a primitive solvable subproblem for the given original problem may gush out the wave with constant amplitude \mathbf{a} . Whereas a wave source node that represents a primitive unsolvable subproblem for the given original problem doesn't generate any wave.

The set of all the discovered output edges of node n by the time t is denoted $\mathcal{Q}_t(n) = \{\langle n, n_1 \rangle, \dots, \langle n, n_q \rangle\}$, where $n_1, \dots, n_q \in N$. The j -th hyper-path $\mathcal{P}_j(n) = \mathcal{L} \sim n$ of node n represents the j -th hyper-edge route from wave sources via hyper-edge $\mathcal{L} \in \Omega(n)$ to node n , where, for any node n_i in $\mathcal{P}_j(n)$, only one input hyper-edge of node n_i belongs to $\mathcal{P}_j(n)$.

Definition 1. The amplitude attenuation $\delta(\mathcal{L} \sim n, t)$ for waves to pass hyper-path $\mathcal{P}(n) = \mathcal{L} \sim n$ and to arrive at node n at time t is defined by

$$\delta(\mathcal{L} \sim n, t) = \sum_{m, \langle m, n \rangle \in \mathcal{L}, \mathcal{L}' \sim m \in \mathcal{P}(n)} [\delta(\mathcal{L}' \sim m, t) + \beta d(m, n)], \quad (1)$$

where $d(m, n)$ represents the cost or distance of edge $\langle m, n \rangle$, and β is a positive proportional coefficient. If n is a wave source, then $\delta(\mathcal{L} \sim n, t) = 0$ at any t .

Definition 2. The wave amplitude along the hyper-path $\mathcal{P}(n) = \mathcal{L} \sim n$ of node n is equal to

$$\alpha(\mathcal{L} \sim n, t) = \begin{cases} \mathbf{a} - \delta(\mathcal{L} \sim n, t) \gg 0, & \text{if } \lambda > 0, \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

where

$$\lambda = \sum_{m, \langle m, n \rangle \in \mathcal{L}, \mathcal{L}' \sim m \in \mathcal{P}(n)} [\alpha(\mathcal{L}' \sim m, t) - \beta d(m, n)] - (p - 1) \mathbf{a}, \quad (3)$$

and p is the number of edges in the hyper-edge \mathcal{L} .

Definition 3. When waves arrive along the hyper-path $\mathcal{P}(m) = \mathcal{L}' \sim m$ up to node m at time $t(\mathcal{P}(m))$, where $\langle m, n \rangle \in \mathcal{L}, \mathcal{L}' \sim m \in \mathcal{P}(n)$, the refraction wave propagates along $\langle m, n \rangle$ to node n after time $t(\mathcal{P}(m))$ at the following speed

$$\mathcal{V}_{\mathcal{P}(m)}(m \sim n, t) = \begin{cases} v_0 d(m, n) / \partial(t) & \text{if } \partial(t) > 0, \\ 0, & \text{if } \partial(t) \leq 0, \end{cases} \quad (4)$$

$$\partial(t) = \beta d(m, n) + h_m(n) - v_0 t(\mathcal{P}(m)) + \delta(\mathcal{P}_t^*(m), t), \quad (5)$$

where v_0 is a positive constant; $h_m(n)$ is the heuristic cost or distance estimated for hereafter propagation from node m via node n to a wave sink \aleph . If node n is a wave sink, then $h_m(n) = 0$. $\delta(\mathcal{P}_t^*(m), t)$ is the smallest attenuation among the waves that arrive at node m along all the hyper-paths, $\forall i, \mathcal{P}_i(m) \in \Omega(m)$, of node m until time t , that is

$$\delta(\mathcal{P}_t^*(m), t) = \min_i \delta(\mathcal{P}_i(m), t) = \min_{k, \langle k, m \rangle \in \mathcal{L}, \mathcal{L} \in \Omega(m)} [\delta(\mathcal{P}_t^*(k), t) + \beta d(k, m)].$$

Definition 4. The time period $T(\mathcal{P}(n))$ for the wave to pass through a hyper-path $\mathcal{P}(n) = \mathcal{L} \sim n$ of node n up to node n is defined by

$$T(\mathcal{P}(n)) = \sum_{m, \langle m, n \rangle \in \mathcal{L}, \mathcal{P}(m) \in \mathcal{P}(n)} [T(\mathcal{P}(m)) + T(m \sim n)], \quad (6)$$

where $T(m \sim n)$ is the period for the wave from $\mathcal{P}(m)$ to propagate from node m to node n .

Lemma 1. *Assume that the waves propagate at the speed given by Eq.(4) and Eq.(5), and $\delta(\mathcal{P}_t^*(m), t)$ doesn't change with the time $t \geq t(\mathcal{P}(m))$, then the period for the wave to pass through the hyper-path $\mathcal{P}(m)$ and the edge $\langle m, n \rangle$ to node n is equal to the period for the wave to pass through the optimal hyper-path $\mathcal{P}^*(m)$ and the edge $\langle m, n \rangle$ to node n , where $\langle m, n \rangle \in \mathcal{L}, \mathcal{L} \in \Omega(n)$.*

Proof. Because $\delta(\mathcal{P}_t^*(m), t)$ still remains constant after $T(\mathcal{P}(m))$, by (4) and (5), $\partial(t)$ remains unchanged and the wave from $\mathcal{P}(m)$ will refract at a constant speed along $\langle m, n \rangle$ at $t \geq t(\mathcal{P}(m))$, whereby

$$\begin{aligned} T(m \sim n) &= d(m, n)/\mathcal{V}_{\mathcal{P}(m)}(m \sim n, t) \\ &= [\delta(\mathcal{P}_t^*(m), t) + \beta d(m, n) + h_m(n)]/v_0 - t(\mathcal{P}(m)), \\ T(\mathcal{P}) &= T(\mathcal{P}(m)) + T(m \sim n) = [\delta(\mathcal{P}_t^*(m), t) + \beta d(m, n) + h_m(n)]/v_0. \end{aligned} \quad (7)$$

Let $\mathcal{P}_{t(\mathcal{P}(m))}^*(m)$ be the optimal hyper-path to node m among all the hyper-paths discovered until $t(\mathcal{P}(m))$. Since $t(\mathcal{P}_{t(\mathcal{P}(m))}^*(m)) \leq t(\mathcal{P}(m))$, the hyper-path with the minimum δ value must be $\mathcal{P}_{t(\mathcal{P}(m))}^*(m)$ during $t(\mathcal{P}_{t(\mathcal{P}(m))}^*(m)) \leq t \leq t(\mathcal{P}(m))$. Thus $\delta(\mathcal{P}_t^*(m), t)$ also doesn't change after $t(\mathcal{P}_{t(\mathcal{P}(m))}^*(m))$, namely Eq.(7) holds true for $\mathcal{P}_{t(\mathcal{P}(m))}^*(m)$. Therefore

$$T(\mathcal{P}(n)) = T(\mathcal{P}_{t(\mathcal{P}(m))}^*(m)) + T(m \sim n).$$

Lemma 2. *Assume that the conditions for Lemma 1 hold true, then the period for waves to propagate along the hyper-path $\mathcal{P}(n) = \mathcal{L} \sim n$ up to node n is equal to that for waves to propagate along the optimal hyper-path up to the input nodes of hyper-edge \mathcal{L} and then to pass via \mathcal{L} up to n .*

Proof. By Eqs.(6) and (7), for the hyper-edge $\mathcal{L} \in \Omega(n)$ we have

$$\begin{aligned} T(\mathcal{P}(n)) &= \sum_{m, \langle m, n \rangle \in \mathcal{L}} [\delta(\mathcal{P}_t^*(m), t) + \beta d(m, n) + h_m(n)]/v_0 \\ &= \sum_{m, \langle m, n \rangle \in \mathcal{L}} [\delta(\mathcal{P}_t^*(m), t) + \beta d(m, n) + h(n)]/v_0 \end{aligned} \quad (8)$$

where $h(n) = \sum_{m, \langle m, n \rangle \in \mathcal{L}} h_m(n)$.

Lemma 3. *Assume that the wave amplitudes conform to Eq.(2) and the conditions for Lemma 1 hold true. If $t^*(m) \geq t^*(n)$, then all the hyper-paths that contain edge $\langle m, n \rangle$ are not optimal to node n , where $t^*(m)$ is the earliest time for waves to arrive at node m .*

Proof. By $t^*(n) = \min_{\mathcal{P}(n)} \{T(\mathcal{P}(n))\}$ and Lemma 2, $t^*(n)$ is equal to the time period for waves to propagate along the optimal hyper-path $\mathcal{P}^*(n)$ to node n . Thus by $t^*(m) \geq t^*(n)$, it may be derived that

$$\begin{aligned} t^*(n) &= T(\mathcal{P}^*(n)) = T(\mathcal{P}_{t^*(n)}^*(n)) = T(\mathcal{P}_{t^*(n)}^*(n)) \\ &= \sum_{k, \langle k, n \rangle \in \mathcal{L}, \mathcal{L} \in \mathcal{P}^*(n)} [\delta(\mathcal{P}_{t^*(k)}^*(k), t^*(k)) + \beta d(k, n) + h_k(n)]/v_0 \\ &= \sum_{k, \langle k, n \rangle \in \mathcal{L}, \mathcal{L} \in \mathcal{P}^*(n)} [t^*(k) + T(k \sim n)] \leq t^*(m) < t^*(m) + T(m \sim n). \end{aligned}$$

Therefore, any hyper-edge that includes the node m and the edge $\langle m, n \rangle$ doesn't belong to the optimal hyper-path $\mathcal{P}^*(n)$ for node n .

Lemma 4. *Assume that the wave amplitudes conform to Eq.(2) and the conditions for Lemma 1 hold true. If $\delta(\mathcal{P}_t^*(m), t)$ never changes for $\forall m$ when $t \geq t^*(m)$, then the wave with the maximum amplitude will arrive earliest at the time $t^*(n) = [\delta(\mathcal{P}_{t^*(n)}^*(n), t^*(n)) + h(n)]/v_0$, where $\langle m, n \rangle \in \mathcal{L}, \mathcal{L} \in \Omega(n)$.*

Proof. By the assumption $\delta(\mathcal{P}_t^*(m), t \geq t_m^*) = \delta(\mathcal{P}_{t_m^*}^*(m), t^*(m)) \equiv c(m)$, where $c(m)$ is a constant related to node m , and by Eq.(8), we have

$$\begin{aligned} t(n) &= T(\mathcal{P}(n)) = \sum_{\langle m, n \rangle \in \mathcal{L}} [\delta(\mathcal{P}_t^*(m), t \geq t^*(m)) + \beta d(m, n)]/v_0 + h(n)/v_0 \\ &= \sum_{\langle m, n \rangle \in \mathcal{L}} [c(m) + \beta d(m, n)]/v_0 + h(n)/v_0, \text{ and} \\ \delta(\mathcal{P}_t^*(n), t \geq t^*(n)) &= \min_{\mathcal{P}(n)} \{ \delta(\mathcal{P}(n), t \geq t^*(n)) \} \\ &= \min_{\mathcal{L} \in \Omega(n)} \{ \sum_{\langle m, n \rangle \in \mathcal{L}} [\delta(\mathcal{P}_t^*(m), t \geq t^*(n)) + \beta d(v, u)] \}. \end{aligned}$$

By Lemma 3, only the case $t^*(m) < t^*(n)$ needs considering, and therefore

$$\begin{aligned} \delta(\mathcal{P}_t^*(n), t \geq t^*(n)) &= \min_{\mathcal{L} \in \Omega(n)} \{ \sum_{\langle m, n \rangle \in \mathcal{L}} [\delta(\mathcal{P}_t^*(m), t > t^*(m)) + \beta d(m, n)] \} \\ &= \min_{\mathcal{L} \in \Omega(n)} \{ \sum_{\langle m, n \rangle \in \mathcal{L}} [c(m) + \beta d(m, n)] \} \\ &= \min_{\mathcal{L} \in \Omega(n)} [v_0 T(\mathcal{P}(n)) - h(n)] = v_0 t^*(n) - h(n), \end{aligned}$$

which implies that $\delta(\mathcal{P}_t^*(n), t)$ is also unchanged for $t \geq t^*(n)$. Moreover, we obtain

$$t^*(n) = [\delta(\mathcal{P}^*(n), t \geq t^*(n)) + h(n)]/v_0 = [\delta(\mathcal{P}_{t^*(n)}^*(n), t^*(n)) + h(n)]/v_0,$$

and the minimum attenuation wave, i.e. the maximum amplitude wave, arrives at n at the time t_u^* earliest.

Lemma 5. *Assume that the wave amplitude and the wave speed conform to Eq.(2) and Eq.(4), respectively. When $t \geq t^*(m)$, it is true that $\delta(\mathcal{P}_t^*(m), t)$ doesn't change with the time for any node m .*

Proof. At first, we define the maximum intermediate node number $\xi(m)$ of node m in a hyper-path $\mathcal{P}_t(m)$ as follows: If node m is a wave source, then $\xi(m) = 0$; If $\max_{h, \langle h, m \rangle \in \mathcal{L}, \mathcal{L} \in \mathcal{P}(m)} \{ \xi(h) \} = k$, then $\xi(m) = k + 1$.

By induction for $\xi(m)$, if $\xi(m) = 0$, it is obvious that $\delta(\mathcal{P}_t^*(m), t) = 0$ for $t \geq 0$ and the Lemma holds true. If $\xi(m) = 1$, then any ancestor node h of m is a wave source, therefore

$$\begin{aligned} \delta(\mathcal{P}_t^*(m), t) &= \min_{\mathcal{L} \in \Omega(m)} \{ \sum_{h, \langle h, m \rangle \in \mathcal{L}, \mathcal{L} \in \mathcal{P}(m)} [\delta(\mathcal{P}_t^*(h), t) + \beta d(h, m)] \} \\ &= \min_{\mathcal{L} \in \Omega(m)} \sum_{h, \langle h, m \rangle \in \mathcal{L}, \mathcal{L} \in \mathcal{P}(m)} \beta d(v, u) \end{aligned}$$

is unchanged with time and the Lemma is true for $\xi(m) = 1$. By the induction assumption for $\xi(m) = k$, one needs proving that the Lemma also holds true for $\xi(m) = k + 1$. For $\xi(m) = k + 1 \geq 2$, the set $\omega(m)$ of the father nodes of node m may be divided into two subsets, $\omega_1(m)$ and $\omega_2(m)$, such that if $h \in \omega(m)$ and $t^*(h) < t^*(m)$ then $h \in \omega_1(m)$, otherwise $h \in \omega_2(m)$. By the induction assumption about $\xi(m) \leq k$ and by Lemma 3, it is sure that $\delta(\mathcal{P}_t^*(h), t \geq t^*(h)) = c(h)$ is a constant. Moreover,

$$\begin{aligned} \delta(\mathcal{P}_t^*(m), t \geq t^*(m)) &= \min_{\mathcal{L}, \mathcal{L} \in \Omega(m)} \{ \sum_{\langle h, m \rangle \in \mathcal{L}} [\delta(\mathcal{P}_t^*(h), t \geq t^*(h)) + \beta d(h, m)] \} \\ &= \min_{\mathcal{L}, \mathcal{L} \in \Omega(m)} \{ \sum_{\langle h, m \rangle \in \mathcal{L}, h \in \omega_1(m)} [c(h) + \beta d(h, m)] \}. \end{aligned}$$

Therefore $\delta(\mathcal{P}_t^*(m), t \geq t^*(m))$ remains unchanged with time, and the conclusion holds for $\xi(m) = k + 1$.

Theorem 1. *Assume that the wave amplitude and the wave speed conform to Eq.(2) and Eq.(4), respectively. Then the wave that propagates along the hyper-path $\mathcal{P}^*(n)$ with the minimum attenuation, i.e. with the maximum amplitude, will arrive at node n earliest, and the arrival time is equal to*

$$t^*(n) = [\delta(\mathcal{P}^*(n), t \geq t^*(n)) + h(n)]/v_0.$$

Proof. It is straightforward from Lemma 3, 4 and 5.

Theorem 2. *Suppose that waves propagate along $\mathcal{P}(m)$ and $\mathcal{P}^*(m)$ to node m at time $t(m)$ and $t^*(m)$, respectively, and $t(m) - t^*(m) < [\beta d(m, n) + h_m(n) - h(m)]/v_0$. Then the waves along hyper-paths, $\mathcal{P}^*(m)$ and $\mathcal{P}^*(m)$, may arrive at the next node n at the same time.*

Proof. By Lemma 4 and Theorem 1, we have

$$t^*(m) = [\delta(\mathcal{P}^*(m), t \geq t^*(m)) + h(m)]/v_0.$$

Then by the assumption $t(m) - t^*(m) < [\beta d(m, n) + h_m(n) - h(m)]/v_0$, it may be derived that

$\beta d(m, n) + h_m(n) - v_0 t(m) + \delta(\mathcal{P}^*(m), t \geq t^*(m)) > 0$, namely by Eq.(5) there is $\partial(t \geq t^*(m)) > 0$. Furthermore, by Definition 3 and Lemma 1, we have $T(\mathcal{P}(m) \sim n) = [\delta(\mathcal{P}^*(m), t \geq t^*(m) + \beta d(m, n) + h_m(n))/v_0 = T(\mathcal{P}^*(m) \sim n)$, which implies that the wave along hyper-paths, $\mathcal{P}^*(m)$ and $\mathcal{P}^*(m)$, may arrive at the next node n at the same time.

On the other hand, if $t(m) - t^*(m) \geq [\beta d(m, n) + h_m(n) - h(m)]/v_0$, then we have $\partial(t \geq t^*(m)) \leq 0$ and $\mathcal{V}_{\mathcal{P}(m)}(m \sim n, t \geq t(\mathcal{P}(m))) = 0$, which means the wave from $\mathcal{P}(m)$ fails to propagate along $m \sim n$.

Theorem 3. *If the heuristic value $h_m(n)$ is selected so that $[h_m(n) - h(m)]/d(m, n)$ is a constant, then the solitary wave propagates at an identical speed along any edge of an infinite implicit AND/OR graph.*

Proof. By Theorem 1, we have $v_0 t^*(m) = \delta(\mathcal{P}^*(m), t \geq t^*(m)) + h_m(n)$. Thus

$$\begin{aligned} & \mathcal{V}_{\mathcal{P}(m)}(m \sim n, t \geq t^*(m)) \\ &= v_0 d(m, n)/[\beta d(v, u) + h_m(n) - v_0 t^*(m) + \delta(\mathcal{P}^*(m), t \geq t^*(m))] \\ &= v_0 d(m, n)/[\beta d(m, n) + h_m(n) - h(m)] = v_0/(\beta + c) \end{aligned}$$

is a positive constant. That is, the wave propagation speed long any edge is identical.

Theorem 4. *The time period spent in finding the optimal solution is independent of selected heuristic values for the same non-object wave node.*

Proof. The optimal solution to search an implicit AND/OR graph means that the optimal hyper-path from wave source nodes up to wave sink nodes. Note that, for any wave sink node n , we have $h(u) \equiv 0$. Hence, by Theorem 1, we have the time $t^*(n) = [\delta(\mathcal{P}_t^*(n), t \geq t^*(n)) + h(n)]/v_0 = \delta(\mathcal{P}_t^*(n), t \geq t^*(n))/v_0$ for wave sink node n . Here $\delta(\mathcal{P}_t^*(n), t \geq t^*(n))$ has nothing to do with the heuristic value $h(m)$ of any intermediate non-object wave node m .

Theorem 5. *The larger the heuristic values, the less the complexity of wave nodes required for finding out the optimal solution.*

Proof. Let the set of nodes via which waves have passed by time t be $\mathcal{N}(t)$, and $h_1(n) \leq h_2(n)$ for wave node n . By Theorem 1, $t_1^*(n) = [\delta(\mathcal{P}_t^*(n), t \geq t_1^*(n)) + h_1(n)]/v_0$ and $t_2^*(n) = [\delta(\mathcal{P}_t^*(n), t \geq t_2^*(n)) + h_2(n)]/v_0$ hold true.

Because $\delta(\mathcal{P}_t^*(n), t \geq t_1^*(n)) = \delta(\mathcal{P}_t^*(n), t \geq t_2^*(n))$, and they are the wave amplitude attenuation along the optimal hyper-path, so that they are irrelative to $h(n)$. Hence we have $t_1^*(n) - h_1(n) = t_2^*(n) - h_2(n)$. Since $h_1(n) \leq h_2(n)$, it leads to $t_2^*(n) \geq t_1^*(n)$, which implies the waves with $h_2(n)$ can't propagate so fast as the waves with $h_1(n)$ can. Moreover, according to Theorem 3, irrespective of $h_1(n)$ or $h_2(n)$, the waves spend the same time in finding the optimal hyper-path, whereby $\mathcal{N}_2(t) \subseteq \mathcal{N}_1(t)$.

Theorem 6. *The existential optimal hyper-path of an infinite implicit AND/OR graph may be obtained by using the solitary wave mechanism Eq.(2) and Eq.(4).*

Proof. By Theorem 1 through Theorem 5, the waves along the optimal hyper-path of a node always propagate at the node earlier than along non-optimal hyper-path of the node. Therefore, as long as there is an object node in a finite distance, even though for an infinite cycle graph, the optimal solution with the shortest distance or cost may be found in a finite time period.

Theorem 7. *For an infinite implicit AND/OR graph, the proposed solitary wave mechanism has the time complexity $O(L)$ to find the existential optimal solution, where L is the cost or distance of the optimal hyper-path.*

Proof. By Theorem 1, the wave reaches a wave sink node n at time $t^*(n)$ which is directly proportional to L .

3 Conclusions

The proposed solitary wave approach and the algorithm may be effectively used in the task allocation and resource assignment in multi-agent systems and computer networks. The solitary wave model is essentially different from searching a general implicit AND/OR graph corresponding to state space. By the solitary wave model, the hyper-edges are dynamically constructed, and the back-tracking and two phases (top-down and bottom-up) are not necessary, so that it is possible to handle the stochastic behaviors in MAS and CN. The theoretical analysis and practical experiments show the advantages of the proposed method in terms of the high-degree parallelism, the suitability for complex environment, and the powerful processing ability.

References

1. Vedantham, S., Iyengar, S.: The Bandwidth Allocation Problem in The ATM Network Model Is NP-Complete. Information Processing Letters 65(4), 179–182 (1998)
2. Huang, Yin-Fu, Chao, Bo-Wei: A Priority-Based Resource Rllocation Strategy in Distributed Computing Networks. Journal of Systems and Software 58, 221–233 (2001)

3. Abdel-Wahab, H., Youssef, A., Maly, K.: Distributed Management of Exclusive Resources in Collaborative Multimedia Systems. In: Proceedings of 3rd IEEE Symposium on Computers and Communications, pp. 115–119. IEEE Press, Los Alamitos (1998)
4. Racz, A., Fias, N., Racz, P.: Effective Bandwidth And Associated CAC Procedure for Traffic Streams Multiplexed over a VBR Link. *Computer Communications* 24, 781–789 (2001)
5. Shuai, D.X., Zhao, H.B.: A New Generalized Cellular Automata Approach to Optimization of Fast Packet Switching. *Computer Networks* 45(4), 399–419 (2004)
6. Thomas, H.Y., Li, B., Shivendra, S.P., Henry, T.: On Network Bandwidth Allocation Policies And Feedback Control Algorithms for Packet Networks. *Computer Networks* 34, 481–501 (2000)
7. Chandramathi, S., Shanmugavel, S.: Fuzzy-Based Dynamic Bandwidth Allocation for Heterogeneous Sources in ATM Network. *Applied Soft Computing Journal* 3(1), 53–70 (2003)
8. Cheng, T.H., Sze, Y.K., Tan, C.W.: A Heuristic Algorithm for Allocating Virtual Path Bandwidth in an ATM Network. *Computer Communications* 22(9), 803–810 (1999)
9. James, E.B., Teunis, J.O., Anthony, E.K., Karen, E.M.: Path Selection And Bandwidth Allocation in MPLS Networks. *Performance Evaluation* 52(2-3), 133–152 (2003)
10. Ahn, K.M., Kim, S.H.: Optimal Bandwidth Allocation for Bandwidth Adaptation in Wireless Multimedia Networks. *Computers and Operations Research* 30(13), 1917–1929 (2003)
11. Han, D.H., Park, C.G., Baik, K.H.: Performance Analysis of an AAL Multiplexer with Dynamic Bandwidth Allocation in an IP/ATM Environment. *Computers and Operations Research* 30(12), 1851–1864 (2003)
12. Kelly, F.P.: *Mathematical Modeling of the Internet*, pp. 685–702. Springer, Berlin (2001)
13. Low, S.H.: A Duality Model of TCP And Queue Management Algorithms. *IEEE ACM Trans. on Networking* 11(4), 525–536 (2003)
14. Low, S.H., Paganini, F., Doyle, J.C.: Internet Congestion Control. *IEEE Control Syst. Mag.* 22(2), 28–43 (2002)
15. Massonlie, L., Roberts, J.: Bandwidth Sharing: Objectives And Algorithms. *IEEE/ACM Transactions on Networking* 10(3), 320–328 (2002)
16. Ito, A., Yano, H.: The Emergence of Cooperation in a Society of Autonomous Agents – The prisoner’s Dilemma Game under the Disclosure of Contract Histories. In: *ICMAS 1995*, San Francisco (1995)
17. Shehory, O., Kraus, S.: Methods for Task Allocation via Agent Coalition Formation. *Artificial Intelligence* 101, 165–200 (1998)
18. Kraus, S., Jonathan, W., Gilad, Z.: Multiagent Negotiation under Time Constraints. *Artificial Intelligence* 75, 297–345 (1995)

A Novel Global Convergence Algorithm: Bee Collecting Pollen Algorithm*

Xueyan Lu^{1,2} and Yongquan Zhou¹

¹ College of Mathematic and Computer Science, Guangxi University for Nationalities
Nanning 530006, Guangxi, China
wonderfullu@163.com

² Department of Computer and Electronic Information Engineer, Wuzhou University,
Wuzhou, 543002, Guangxi, China
yongquanzhou@126.com

Abstract. Inspired by the behavior of the honeybees' collecting pollen, Bee Collecting Pollen Algorithm (BCPA) is proposed in this paper. This is a novel global convergence searching algorithm. It simulates the behavior of the honeybees' collecting pollen and describes the swarm intelligent. The experiment for TSP shows that the improve algorithm is more efficient.

Keywords: Honeybee, collect pollen, swarm intelligent, honey resource, global convergence.

1 Introduction

Swarm intelligence has gained increasingly high interest among the researchers from different areas, like, science, commerce and engineering over the last few years. It is particularly suitable to apply methods inspired by swarm intelligence to various optimization problems, especially if the space to be explored is large and complex.

Swarm Intelligence is an emerging field of artificial intelligence. It is concerned with modeling of social interactions between social beings, primarily ants, birds and, in the recent time, bees. This approach utilizes simple and flexible agents that form a collective intelligent as a group. This is an alternate approach to traditional intelligence models, exhibiting features of autonomy, emergence, robustness and self-organization. Several examples of artificial models inspired by interactions of social organisms are: Particle swarm optimization (PSO) methods, a population based stochastic optimization technique developed in 1995, by Eberhard and Kennedy [1]. It is inspired by flocking behavior of the birds searching for food. Although PSO methods share many common attributes with GA, such as stochastic nature, population of solution candidates, PSO methods, unlike GA use a kind of cooperation between particles to drive the search process. PSO methods have no evolutionary operators like crossover and mutation. Each particle keeps track of its own best solution, and the best solution found so far by the swarm. It means the particles possess own and

* This work was supported by Grants 60461001 from NSF of China and project supported by grants 0542048; 0832082 from Guangxi science foundation, and Innovation Project of Guangxi Graduate Education.

collective memory, and are able to communicate. The difference between global best and personal best is used to direct particles in search space. Another model of swarm-based approaches to optimization is Ant Colony Optimization (ACO), where the search process is inspired by the collective behavior of trail deposit and follow-up, observed by real ant colonies. A colony of simple (trial of pheromones) and thus proposes solution to a problem, based on their collective experience. Ant colony algorithms as evolutionary optimization algorithms were first proposed by Dorigo (1992) [2] as a multiagent approach to different combinatorial optimization problems like traveling sales man problem and the quadratic assignment problem. Although, the first result were not very encouraging, it initiated the interest among research community, and since then, several algorithms have been proposed, some of them showing very convincing results.

A Classical Traveling Salesman Problem (TSP) has been an interesting problem. Given a number of nodes and their distances of each other, an optimal travel route is to be calculated so that starting from a node and visit every other node only once with the total distance covered minimized. As a special optimization problem, there are many people have been do much work on it, i.e. Neural Network, Potential Fields, Genetic algorithms, Particles Swarm Optimization methods and other. Optimization methods can generally be classified into two distinct groups: direct search and gradient based search. Gradient based search techniques require derivative information of the function and constraints, while direct search methods use only objective function and constraint values. Since derivative information is not used, these methods generally require large number of iterations for convergence, but are at the other hand applicable to very broad problem space. However, TSP is inherently a NP hard problem with no easy solution.

The behavior of honey-bees shows many features like cooperation and communication, so honey-bees have aroused great interests in modeling intelligent behavior these years [3]-[19], but these algorithms are most mechanism by the marriage in bee. This paper describes a novel approach that uses the honey bees foraging model to solve the problem. Experimental results comparing the proposed honey bee colony approach with existing approaches such as ant colony and Genetic Algorithm (GA) will be presented.

This paper first describes how honey bee colonies deploy forager bees to collect nectar amongst diverse flower patches. The mapping of Traveling salesman problem to honey bee's forager deployment is given. Subsequently, the implementation details are discussed in section 3. This is then followed by a comparative study on the performance of the honey bee approach on benchmark problems in section 4. The paper finally ends with conclusions and future works in Section 5.

2 Honey Bee Colony

Colonies of social insects such as ants and bees have instinct ability known as swarm intelligence. This highly organized behavior enables the colonies of insects to solve problems beyond capability of individual members by functioning collectively and interacting primitively amongst members of the group. In a honey bee colony for example, this behavior allows honey bees to explore the environment in search of flower patches (food sources) and then indicate the food source to the other bees of

the colony when they return to the hive. Such a colony is characterized by self organization, adaptiveness and robustness.

Seeley (1995) proposed a behavioral model of self organization for a colony of honey bees. In the model, foraging bees visiting flower patches return to the hive with nectar as well as a profitability rating of respective patches. The collected nectar provides feedback on the current status of nectar flow into the hive. The profitability rating is a function of nectar quality, nectar bounty and distance from the hive. The feedback sets a response threshold for an enlisting signal which is known as waggle dance, the length of which is dependent on both the response threshold and the profitability rating. The waggle dance is performed on the dance floor where individual foragers can observe. The foragers can randomly select a dance to observe and follow from which they can learn the location of the flower patch and leave the hive to forage. This self organized model enables proportionate feedback on goodness of food sources [14].

3 Bee Collecting Pollen Algorithm

This section details algorithms to perform TSP inspired by the behavior of honey bee colony. The challenge is to adapt the self-organization behavior of the colony for solving TSP. There are two major characteristics of the bee colony in searching for food sources: waggle dance and forage (or nectar exploration). We will discuss in separate sub-sections on how we map these characteristics of a bee colony to TSP.

3.1 Waggle Dance

A forager f_i on return to the hive from nectar exploration will attempt with probability p to perform waggle dance on the dance floor with duration $D = d_i A$, where d_i changes with profitability rating while A denotes waggle dance scaling factor. Further, it will also attempt with probability r_i to observe and follow a randomly selected dance. The probability r_i is dynamic and also changes with profitability rating. If a forager chooses to follow a selected dance, it will use the ‘path’ taken by the forager performing the dance to guide its direction for flower patches. We term the path as ‘preferred path’. The path for a forager is a series of landmarks from a source (hive) to a destination (nectar). For TSP, the profitability rating should be related to the objective function, which in our case, is make span. Let Pf_i denote the profitability rating for a forager, it is given by:

$$Pf_i = \frac{1}{C_{\max}^i} \quad (1)$$

where, C_{\max}^i = make span of the schedule generated by a forager f_i . The bee colony’s average profitability rating, Pf_{colony} is given by:

$$Pf_{colony} = \frac{1}{n} \sum_{j=1}^n \frac{1}{C_{\max}^j} \quad (2)$$

where, n = number of waggle dance at time t (we only consider those bees that dance when computing profitability rating); C_{\max}^j = make span of the schedule generated by a forager f_j performing waggle dance. The dance duration, d_i is given by:

$$d_i = \frac{P f_i}{P f_{colony}} \tag{3}$$

The probability r_i of following a path is adjusted according the profitability ratings of a forager and the colony based on the lookup table 1 (adopted from Nakrani and Tovey 2004). Essentially, a forager is more likely to randomly observe and follow a waggle dance on the dance floor if its profitability rating is low as compared to the colony's.

Table 1. Look up Table for Adjusting Probability of Following a Waggle Dance

Profitability Rating	r_i
$P f_i < 0.9 P f_{colony}$	0.60
$0.9 P f_{colony} \leq P f_i < 0.95 P f_{colony}$	0.20
$0.95 P f_{colony} \leq P f_i < 1.15 P f_{colony}$	0.02
$1.15 P f_{colony} \leq P f_i$	0.00

3.2 Forage (Nectar Exploration)

For foraging algorithm, a population of l foragers is defined in the colony. These foragers cyclically construct solutions to the TSP. The foragers move along branches from one node to another node in the disjunctive graph and so construct paths representing solutions. A forager must visit every node once and only once in the graph, starting from initial node (i.e. source) and finishing at final node (i.e. sink), so as to construct a complete solution. When a forager is at a specific node, it can only move to next node that is defined in a list of presently allowed nodes, imposed by precedence constraints of operations. A forager chooses the next node from the list according to the state transition rule:

$$P_{ij}(t) = \frac{[\rho_{ij}(t)]^\alpha \cdot \left[\frac{1}{d_{ij}}\right]^\beta}{\sum_{j \in allowed_nodes} [\rho_{ij}(t)]^\alpha \cdot \left[\frac{1}{d_{ij}}\right]^\beta} \tag{4}$$

where, ρ_{ij} = rating of the edge between $node_i$ and $node_j$; d_{ij} = heuristic distance between $node_i$ and $node_j$; P_{ij} = probability to branch from $node_i$ and $node_j$;

The rating ρ_{ij} of the edge (directed) between $node_i$ and $node_j$ is given by:

$$\rho_{ij} = \begin{cases} \alpha \\ \frac{1 - m \alpha}{k - m} \end{cases}$$

where, α = value assigned to the preferred path, ($\alpha < 1.0$); k = number of allowed nodes; m = number of preferred path, $m = 1$ or 0 ; Based on the expression, it should be noted that for the first nectar exploration expedition by the foragers, ρ_{ij} will be assigned the same value for all allowed nodes (since $m = 0$). The parameters α and β tune the relative importance in probability of the 'weight' in edges found in the preferred path versus the heuristic distance. According to this rule, edges that are found in the preferred path and that are shorter will have a higher probability to be chosen for the solution. The heuristic distance is the processing time of the operation associated with *node j*. When a forager completes a full path, the edges it has traveled and the make span of the resulting solution will be kept for the waggle dance when it returns to the hive.

3.3 The Disturbance from the Environment

There are many factors from the environment that influenced the bee's collect pollen. Meanwhile, when the best value keep on very little change, the algorithm should has gotten stuck in a local optimum to a certain degree. So there is a certain control in the algorithm. The bee's position will be changed by itself, given by added a random number.

3.4 The Death

A honey bee has only 6 weeks life. And in order to avoid premature, to stimulate the circle of individual turn the old and new and to reinforce the exploitation, a new population of bees will be introduced in each generation which could extend the search area in the algorithm.

3.5 The Bulletin

The Bulletin is used to record the best choice of all the bees. Each bee which has changed the position will compare with the bulletin. If the change one is better than the bulletin, the bulletin will be correct. This way can lead to the global search.

3.6 The Ending of the Algorithm

As an iterative algorithm, a good guideline of ending is needed. The max generation of 5 thousand is used to control the circle in the improved algorithm.

3.7 Algorithmic Framework

A combination of forage and waggle dance algorithms constitutes one cycle (or iteration) in this evolutionary computation approach. This computation will run for a

specific number of iterations N_{\max} . The best solution during the iteration process will be presented as final schedule at the end of run. The algorithmic framework of the scheduling algorithm is presented in Algorithm 1.

Algorithm 1. Algorithmic framework for TSP

```

for i = 1 to  $N_{\max}$ 
  for j = 1 to l
    Forage
    Save best solution
    Waggle dance
  if change little
    new population
  end if
end for
end for

```

4 Simulation

To test the global convergence searching performance of the algorithm, we use the TSP. Here TSP based on the data from TSPLIB is GA algorithm respectively. The results are showed by Figure 1 to Figure 4.

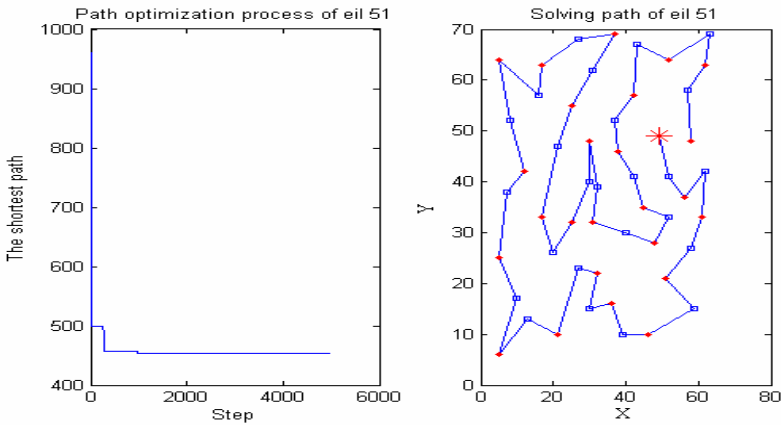


Fig. 1. Path optimization process of eil 51

Fig. 2. Solving path of eil 51 nodes

(the best value as far is 426, our result is 434.3195)

Since the improved algorithm has great randomness, every parameter will be established by many times trial. By the experiment of NP hard problem TSP, improvement of the convergence rate has been greatly changed. Lower than 1 thousand generations, the improved algorithm will reach the global convergence point and the CPU time is less.

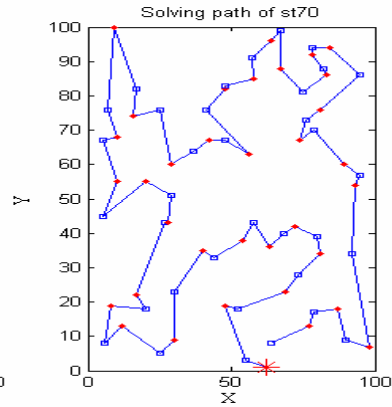
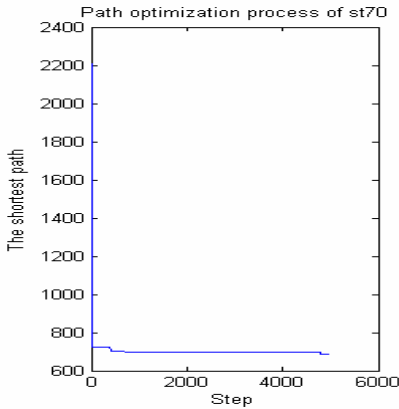


Fig. 3. Path optimization process of st70 **Fig. 4.** Solving path of st70 nodes

(the best value is 675, our result is 695.4688)

5 Conclusions

A novel global convergence searching algorithm based on self-organization of honey bee colony has been implemented for solving TSP. It is found that the performance of the algorithm is comparable to ant colony algorithms, but gaps behind the efficient tabu search heuristics. Since the bee algorithm is our first implementation, we believe there is much room for improvement. We intend to test the bee algorithms on TSP. One of the works we intend to pursue is to deploy the algorithms in a distributed computing environment using software agents. Prior work has already been carried out using agents in symbiotic simulation of semiconductor assembly and test operation (Low et. al. 2005). In comparison to ant colony algorithms, it will be relatively easier to treat each bee forager as an agent in the bee colony algorithms since the issue of share state in maintaining a global pheromone table in ant algorithms does not occur.

References

1. Eberhart, R.C., Kennedy, J.: A New Optimizer Using Particle Swarm Theory. In: Proceedings of the Sixth International Symposium on Micromachine and Human Science, Nagoya, Japan, pp. 39–43 (1995)
2. Dorigo, M., Maniezzo, V., Colomi, A.: Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* 26(1), 29–41 (1996)
3. Abbass, H.A.: Marriage in Honeybees Optimization (MBO): A Haplometrosis Polygynous Swarming Approach. In: The Congress on Evolutionary Computation, Seoul, Korea. CEC 2001, pp. 207–214 (2001)

4. Abbass, H.A.: Amonogenous MBO Approach to Satisfiability. In: The International Conference on Computational Intelligence for Modelling, Control and Automation, CIMCA 2001, LasVegas, NV, USA (2001)
5. Abbass, H.A., Teo, J.: A True Annealing Approach to the Marriage in Honeybee's Optimization Algorithm. In: Proc. of the Inaugural Workshop on Artificial Life (AL 2001), pp. 1–14 (2001)
6. Bozorg, H.O., Afshar, A.: MBO (Marriage Bees Optimization): A New Heuristic Approach in Hydro Systems Design and Operation. In: Proceedings of 1st International Conference on Managing Rivers in the 21st Century: Issues and Challenges, Penang, Malaysia, pp. 499–504 (2004)
7. Bozorg, H.O., Afshar, A., Marin, M.A.: Honeybees Mating Optimization Algorithm (HBMO): A New Heuristic Approach for Engineering Optimization. In: Proceeding of the First International Conference on Modeling, Simulation and Applied Optimization (ICMSA 2005), Sharjah, UAS (2005)
8. Afshar, A., Bozorg, H.O., Marin, M.A., Adams, B.J.: Honeybee Mating Optimization (HBMO) Algorithm for Optimal Reservoir Operation. Journal of the Franklin Institute (in Press, 2006)
9. Nakrani, S., Tovey, C.: On Honey Bees and Dynamic Allocation in an Internet Server Colony. *Adaptive Behavior* 12(3-4), 223–240 (2004)
10. Teodorovic, D., Dell'orco, M.: Bee Colony Optimization: A Cooperative Learning Approach to Complex Transportation Problems. *Advanced OR and AI Methods in Transportation*, 51–60 (2005)
11. Sung, H.J.: Queen-Bee Evolution for Genetic Algorithms. *Electronics Letters* 39(6), 575–576 (2003)
12. Mohammad, F.B.A.: A Honeybee-Mating Approach for Cluster Analysis. Springer, London (2007)
13. Bozorg, H.O., Afshar, A., Marin, M.A.: Honey-Bees Mating Optimization Algorithm (HBMO): A New Heuristic Approach for Water Resource Optimization. *Water Resource Management* 20, 661–680 (2006)
14. Hyeong, S.C.: Converging Marriage in Honey-Bees Optimization and Application to Stochastic Dynamic Program. *Journal of Global Optimization* 35, 423–441 (2006)
15. Chin, S.C., Malcolm, Y.H.L., Appa, I.S., Kheng, L.G.: A Bee Colony Optimization Algorithm to Job Shop Scheduling. In: Proceeding of the 2006 Winter Simulation Conference, pp. 1954–1961 (2006)
16. Curkovic, P., Jerbic, B.: Honey-Bees Optimization Algorithm Applied to Path Planning Problem. *Int. J. Simul. Model* 6(3), 154–164 (2007)
17. Abbass, H.A.: A Single Queen Single Worker Honey-bees Approach to 3-SAT. In: Proceedings of the Genetic and Evolutionary Computation Conference, San Francisco, USA, pp. 807–814 (2001)
18. Teo, J., Abbass, H.A.: An Annealing Approach to the Mating-flight Trajectories in the Marriage in Honey Bees Optimization Algorithm, Technical Report CS04/01, School of Computer Science, University of New South Wales at ADFA (2001)
19. Seeley, T.D.: *The Wisdom of the Hive*. Publication Harward University Press
20. <http://elib.zib.de/pub/Packages/mp2testdata/tsp/tsplib/tsplib.html>

An Enhanced Swarm Intelligence Clustering-Based RBF Neural Network Detection Classifier

Yong Feng, Zhong-fu Wu, Jiang Zhong, Chun-xiao Ye, and Kai-gui Wu

College of Computer Science, Chongqing University, Chongqing, 400030, China
fengyong@cqu.edu.cn

Abstract. The central problem in training a radial basis function neural network (RBFNN) is the selection of hidden layer neurons, which includes the selection of the center and width of those neurons. In this paper, we propose an enhanced swarm intelligence clustering (ESIC) method to select hidden layer neurons, and then, training a cosine RBFNN base on gradient descent learning process. Also, the new method is applied for intrusion detection. Experimental results show that the average DR and FPR of our ESIC-based RBFNN detection classifier maintained a better performance than BP, SVM and OLS RBF.

Keywords: swarm intelligence, clustering, radial basis function neural network, intrusion detection, classifier.

1 Introduction

The relationship between the performance of RBFNNs and their size motivated the development of network construction and/or pruning procedures for autonomously selecting the number of RBFs [1]. The problems of determining the number, shapes, and locations of the RBFs are essentially related to and interact with each other. Solving these problems simultaneously was attempted by developing a multi-objective evolutionary algorithm [2]. In this paper, we focus on new clustering method based on Swarm Intelligence Clustering (SIC) [3, 4] and applied it to the construction of the hidden layer of RBFN.

The rest of the paper is organized as follows. In Section 2, our enhanced swarm intelligence clustering (ESIC) algorithm is developed to get candidate hidden neurons, and the ESIC-based cosine RBF neural network training process is presented. Experiment results are reported in Section 3 and some conclusions are also provided towards the end.

2 ESIC-Based RBFNN

Because the activated function of RBF network is local, the nearer the input sample closes to the center of the kernel function, the greater output value the hidden node produces. Therefore, it is very important to choose an accurate the center of the kernel function for improving the efficiency of RBF network.

2.1 Choosing the Center of the Radial Basis Function of RBFNN Using ESIC

This paper presents a self-adaptive clustering algorithm to determine the center of the radial basis function based on our enhanced swarm intelligence clustering (ESIC) algorithm.

Def.1. Swarm similarity is the integrated similarity of a data object with other data objects within its neighborhood.

A basic formula of measuring the swarm similarity is showed as formula (1).

$$f(o_i) = \sum_{o_j \in Neigh(r)} \left| 1 - \frac{d(o_i, o_j)}{\beta} \right|. \quad (1)$$

Where $Neigh(r)$ denotes the local region, it is usually a rounded area with a radius r , $d(o_i, o_j)$ denotes the distance of data object o_i with o_j in the space of attributes. The parameter β is defined as swarm similarity coefficient.

Def.2. Probability conversion function is a function of $f(o_i)$ that converts the swarm similarity of a data object into picking-up or dropping probability for a simple agent.

The picking-up probability and the dropping probability are given by::

$$P_p = \frac{1}{2} - \frac{1}{\pi} \arctan \left[\frac{f(o_i)}{\alpha} \right]. \quad (2)$$

$$P_d = \frac{1}{2} + \frac{1}{\pi} \arctan \left[\frac{f(o_i)}{\alpha} \right]. \quad (3)$$

Where α is a plus constant and can speed up the algorithm convergence if it is decreased [5].

The **ESIC** algorithm is described as follows:

Input: sample modes X

Output: the candidate hidden neurons H (clusters), k is the pattern clustering number, c_j ($j = 1, 2, \dots, k$) is the clustering center.

- 1) Initialize β , *ant-number*, maximum iterative times n , α , and other parameters.
- 2) Project the data objects on a plane at random, i.e. randomly give a pair of coordinate (x, y) to each data object.
- 3) Give each ant initial objects, initial state of each ant is unloaded.
- 4) for $i=1, 2, \dots, n$ //while not satisfying stop criteria
 - for $j=1, 2, \dots, \textit{ant-number}$
 - a) Compute $f(o_i)$ within a local region with radius r by formula (1).
 - b) If the ant is unloaded, compute P_p by formula (2). Compare P_p with a random probability P_r , if $P_p < P_r$, the ant does not pick up this object, another data object is randomly given the ant, else the ant pick up this object, the state of the ant is changed to loaded.

- c) If the ant is loaded, compute P_d by formula (3). Compare P_d with P_r , if $P_d > P_r$, the ant drops the object, the pair of coordinate of the ant is given to the object, the state of the ant is changed to unloaded, another data object is randomly given the ant, else the ant continue moving loaded with the object.
- 5) for $i = 1, 2 \dots \dots \text{pattern-num}$ //for all patterns
 - a) If an object is isolated, that is the number of its neighbor is less than a given constant, than label it as an outlier.
 - b) Else label this pattern a cluster serial number; recursively label the same serial number to those patterns whose distance to this pattern is smaller than a short distance $dist$. i.e. collect the patterns belong to a same cluster on the agent-work plane.
 - c) Serial number $serial-num++$.
- 6) Compute the cluster means of the $serial-num$ clusters as initial cluster centers.
- 7) Repeat
 - a) (Re)Assign each pattern to the cluster to which the pattern is the most similar, based on the mean value of the patterns in the cluster.
 - b) Update the cluster means, i.e. calculate the mean value of the patterns for each cluster.
- 8) Until no change.

2.2 Training Cosine RBFNN

After choosing the center of the RBFs, we can employ gradient descent learning process to training a cosine RBFNN and remove some redundant neurons.

Consider an RBFNN with inputs from R^m , c RBFs and K output units. Let $v_j \in R^m$ be the prototype that is center of the j th RBF and $w_i = [w_{i1}, w_{i2}, \dots, w_{ic}]^T$ be the vector containing the weights that connect the i th output unit to the RBFs. Define the sets $V = \{v_i\}$ and $W = \{w_j\}$ and let also $A = \{a_i\}$ be a set of free parameters associated with the RBFs. An RBFNN is defined as the function $N : R^m \rightarrow R^n$ that maps $\mathbf{x} \in R^m$ to $N(V, W, A; \mathbf{x})$, such that:

$$\prod_i N(V, W, A; \mathbf{x}) = f\left(\sum_{j=1}^c w_{ij} g_j(\|\mathbf{x} - v_j\|^2) + w_{i0}\right). \quad (4)$$

Where $f(x) = 1/(1 + e^{-x})$ used in this paper, g_j is represents the response of the RBF centered at the prototype v_j . Using this notation, the response of the i th output unit to the input x_k is:

$$\tilde{y}_{i,k} = \prod_i N(x_k) = f\left(\sum_{j=1}^c w_{ij} g_{j,k} + w_{i0}\right). \quad (5)$$

Where $g_{j,k}$ represents the response of the RBF centered at the prototype v_j to the input vector x_k . Unlike the traditional RBFNN using the exponential functions, in this paper, we using a cosine function [5] for $g_{j,k}$ is:

$$g_{i,k} = a_j / (\|x_k - v_j\|^2 + a_j^2)^{1/2}. \tag{6}$$

Cosine RBFNNs can be trained by the original learning algorithm, which was developed by using “stochastic” gradient descent to minimize:

$$E_k = 1 / 2 \sum_{i=1}^n (\tilde{y}_{i,k} - y_{i,k})^2, \quad k = 1, 2, \dots, M. \tag{7}$$

For sufficiently small values of the learning rate, sequential minimization of E_k , leads to a minimum of the total error $E = \sum_{k=1}^m E_k$. After an example (x_k, y_k) is presented to the RBFNN, the new estimate $w_{i,k}$ of each weight vector w_i , is obtained by incrementing its current estimate by the amount $\Delta w_{i,k} = -\beta \nabla_{w_i} E_k$, where ξ is the learning rate.

$$w_{i,k} = w_{i,k-1} + \Delta w_{i,k} = w_{i,k-1} + \xi g_{i,k} \tilde{y}_{i,k} (1 - \tilde{y}_{i,k}) (y_{i,k} - \tilde{y}_{i,k}). \tag{8}$$

The new estimate $a_{j,k}$ of each reference distance a_j , can be obtained by incrementing its current estimate by the amount as $\Delta a_{j,k} = -\xi \partial E_k / \partial a_j$.

$$\begin{aligned} a_{i,k} &= a_{i,k-1} + \Delta a_{i,k} = a_{i,k-1} + \xi g_{j,k} (1 - g_{j,k}^2) \mathcal{E}_{j,k}^h / a_{i,k-1} \\ \mathcal{E}_{j,k}^h &= (g_{j,k}^3 / a_j^2) \sum_{i=1}^c f'(\tilde{y}_{i,k}) (y_{i,k} - \tilde{y}_{i,k}) w_{i,j} \end{aligned} \tag{9}$$

According to (6), the j th cosine RBF can be eliminated during the training process if its reference distance a_j approaches zero.

2.3 ESIC-Based RBFNN Classifier

ESIC-based RBFNN classifier includes training and testing processes can be shown as Fig. 1.

We can get new algorithm to training RBF classifier which uses ESIC algorithm to get the candidate hidden neurons firstly, and then, training the neural network base on gradient descent learning process descried in this section.

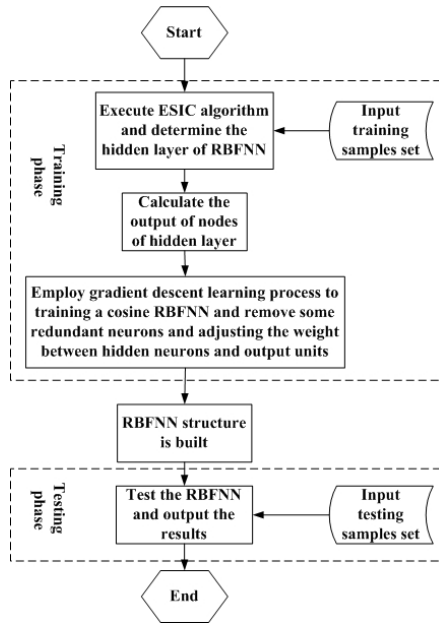


Fig. 1. ESIC-based RBFNN classifier

3 Experiment

The data for our experiments was prepared by the 1998 DARPA intrusion detection evaluation program by MIT Lincoln Labs [6]. The dataset contains 24 attack types that could be classified into four main categories attack types fall into four main categories: DoS, R2L, U2R and Probe.

Several different kinds of classification methods are compared with our ESIC-based RBFNN classifier on DARPA. In the experiment, we compare our approach with the traditional OLS RBF classifier, SVM (Support Vector Machines) classifier and BP neural network classifier.

Five testing data sets (TDS1-TDS5) are set up to perform the algorithm. Each data set contains 6,000 instances (consisted of 1% to 1.5% intrusion instances and 98.5% to 99% normal instances), all of which are selected at random from the normal data set and abnormal data set respectively. Fig. 2 shows attacks number in the data sets.

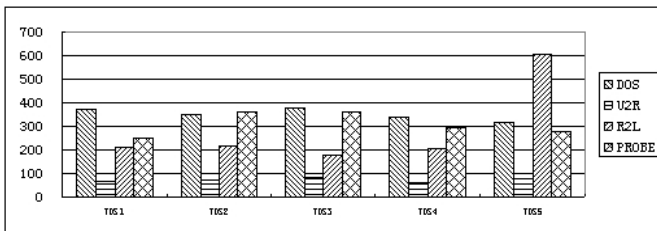


Fig. 2. Attacks number in the data sets

To evaluate the algorithm we are interested in two major indicators of performance: DR (Detection Rate) and FPR (False Positive Rate).

Table 1. Experimental results for ESIC-based RBFNN classifier

β (Swarm Similarity Coefficient for ESIC)	Initial RDFs Count	Final RDFs Count	DR (%)	FPR (%)
0.85	92	61	97.13	0.39
0.80	91	61	97.15	0.37
0.75	87	58	97.59	0.28
0.70	85	55	97.56	0.24
0.65	81	54	97.77	0.24
0.60	74	53	98.21	0.20
0.55	69	52	98.17	0.19
0.50	64	52	98.36	0.17
0.45	60	52	98.45	0.18
0.40	55	51	98.77	0.15
0.35	52	50	98.83	0.13
0.30	50	47	99.01	0.10
0.25	47	46	99.12	0.09
0.20	46	45	99.34	0.08

The BP algorithm were trained with n_h hidden neurons, n_h was varied from 10 to 50, and the maximum training cycles is 5,000. The width parameter of radial function is the most important to the OLS RBF classifier; it varied from 1 and 4 in the paper and the maximum RBFs is 100. In SVM classifier experiment, the kernel function is $K(u, v) = (u \cdot v + 1)^d$, where d is the order of the kernel function, the value of d is bigger and the performance of SVM classifier is higher, so we adjust d from 1 to 5 (interval is 1). For the ESIC-based RBFNN classifier, the learning rate used for updating the output weights and prototypes of the cosine RBFNNs was $\xi = 0.01$, the swarm similarity coefficient β from 0.85 to 0.20 (interval is 0.05), the other parameters are set as follows: *ant-number* = 10, $r = 12$, $\alpha = 10$, $n = 60 \times 1000$. The experimental results for ESIC-based RBFNN classifier are reported in Table 1.

The results from the table 1 shows that, in despite of the variant of swarm similarity coefficient influence the number of the original RBFNNs, this classifier keeps a good performance that the average DR is higher than 98.24% and the FPR is lower than 0.21%. Also it is showed that the false positive is slight down with swarm similarity coefficient become smaller, and it means that ESIC could find the better prototypes for the dataset with a longer training process.

According to testing results, we found that, the BP network and the SVM have the better results than ESIC-based RBFNN at the most time for DR (BP is higher than 98.57% and SVM is higher than 98.97%). However, the network structure of BP neural network is difficult to be determined for the higher dimensional pattern classification problems and cannot be proved to converge well. Also it found that the

Table 2. Experimental results for BP classifier

hidden neurons n_h	DR (%)	FPR (%)
10	98.02	0.51
20	98.10	0.43
30	98.13	0.38
35	98.19	0.37
40	99.18	0.33
45	99.21	0.29
50	99.20	0.30

Table 3. Experimental results for SVM classifier

d (the order of the kernel function)	DR (%)	FPR (%)
1	98.48	0.40
2	99.10	0.33
3	99.23	0.26
4	99.41	0.12
5	98.63	0.34

Table 4. Experimental results for OLS RBF classifier

δ (width parameter for OLS RBF)	DR (%)	FPR (%)
1.0	85.74	0.26
1.5	85.89	0.37
2.0	87.11	0.61
2.5	91.22	0.82
3.0	92.34	0.72
3.5	94.72	0.77
4.0	96.05	0.80

detection rate of ESIC-based RBFNN classifier is increased obviously than the traditional RBF network classifier. The FPR is lower also lower than BP (lower than 0.37%), SVM (lower than 0.29%), and OLS RBF (lower than 0.62%).

4 Conclusions

This paper propose an enhanced swarm intelligence clustering-based RBF neural network classifier, which contains two main stages: employing ESIC to find the candidate hidden neurons; and then use some removing criterion to delete the redundant neurons and adjusting the weight between hidden neurons and output units. Experimental results indicate that our ESIC-based RBFNN classifier has the best detection ability for the network intrusion detection when compared with other conventional classifiers for our tested pattern classification problems.

Acknowledgement. This work is supported by the Natural Science Foundation Project of CQ CSTC (Grant No. 2007BB2192) and National "11th Five-Year Plan" Major Science and Technology Research Project of China (Grant No. 2006BAH02A24-6).

References

1. Mao, K.Z., Huang, G.B.: Neuron Selection for RBF Neural Network Classifier Based on Data Structure Preserving Criterion. In: IEEE Trans. Neural Networks, pp. 1531–1540. IEEE Computational Intelligence Society, New York (2005)
2. Gonzalez, J., Rojas, I., Ortega, J.: Multiobjective Evolutionary Optimization of the Size, Shape, and Position Parameters of Radial Basis Function Networks for Function Approximation. In: IEEE Trans. Neural Networks, pp. 1478–1495. IEEE Computational Intelligence Society, New York (2003)
3. Han, Y.F., Shi, P.F.: An Improved Ant Colony Algorithm for Fuzzy Clustering in Image Segmentation. *Neurocomputing* 70, 665–671 (2007)
4. Runkler, T.A.: Ant Colony Optimization of Clustering Models. *International Journal of Intelligent Systems* 20, 1233–1251 (2005)
5. Feng, Y., Zhong, J., Xiong, Z.Y., Ye, C.X., Wu, K.G.: Network Anomaly Detection Based on DSOM and ACO Clustering. In: Liu, D., Fei, S., Hou, Z., Zhang, H., Sun, C. (eds.) *ISNN 2007. LNCS*, vol. 4492, pp. 947–955. Springer, Heidelberg (2007)
6. MIT Lincoln Laboratory (1998), <http://www.ll.mit.edu/IST/>

Application of the Particle Filter for Simple Gesture Recognition

Yang Weon Lee

Department of Information and Communication Engineering, Honam University,
Seobongdong, Gwangsan-gu, Gwangju, 506-714, South Korea

Abstract. This paper describes a gesture recognition algorithm based on the particle filters, namely CONDENSATION. The particle filter is more efficient than any other tracking algorithm because the tracking mechanism follows Bayesian estimation rule of conditional probability propagation.

1 Introduction

In this paper, we focused into the development of human gesture recognition using particle filter. Particle filter [1] is based on the Bayesian conditional probability such as *prior* distribution and *posterior* distribution. First of all, we expanded the existing algorithm [2] to derive the CONDENSATION-based particle filter for human gesture recognition. Also, we adopt the two hand motion model to confirm the algorithm performance such as leftover and paddle. The overall scheme for the gesture recognition system is shown in Figure 1.

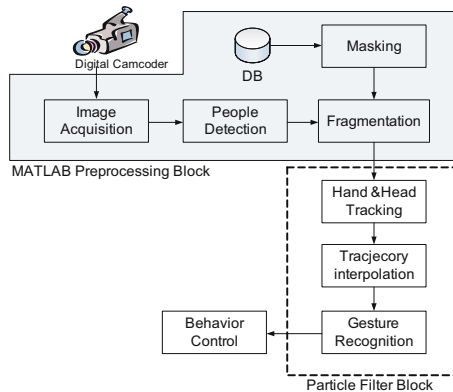


Fig. 1. Overall operation block diagram of recognition system

2 Condensation Algorithm

2.1 Condensation Algorithm

The particle filter approach to track motion, also known as the condensation algorithm [1] and Monte Carlo localisation, uses a large number of particles to

explore the state space. Each particle represents a hypothesised target location in state space. Initially the particles are uniformly randomly distributed across the state space, and each subsequent frame the algorithm cycles through the steps as follows:

1. Deterministic drift: particles are moved according to a deterministic motion model (a damped constant velocity motion model was used).
2. Update probability density function (PDF): Determine the probability for every new particle location.
3. Resample particles: 90with replacement, such that the probability of choosing a particular sample is equal to the PDF at that point; the remaining 10throughout the state space.
4. Diffuse particles: particles are moved a small distance in state space under Brownian motion.

This results in particles congregating in regions of high probability and dispersing from other regions, thus the particle density indicates the most likely target states. See [3] for a comprehensive discussion of this method. The key strengths of the particle filter approach to localisation and tracking are its scalability (computational requirement varies linearly with the number of particles), and its ability to deal with multiple hypotheses (and thus more readily recover from tracking errors). However, the particle filter was applied here for several additional reasons:

- it provides an efficient means of searching for a target in a multi-dimensional state space.
- reduces the search problem to a verification problem, ie. is a given hypothesis face-like according to the sensor information?
- allows fusion of cues running at different frequencies.

2.2 Application of Condensation for the Gesture Recognition

In order to apply the Condensation Algorithm to gesture recognition, we extend the methods described by Black and Jepson [2]. Specifically, a state at time t is described as a parameter vector: $s_t = (\mu, \phi^i, \alpha^i, \rho^i)$ where: μ is the integer index of the predictive model, ϕ^i indicates the current position in the model, α^i refers to an amplitudal scaling factor and ρ^i is a scale factor in the time dimension. Note that i indicates which hand's motion trajectory this ϕ^* , α^* , or ρ^* refers to left and right hand where $i \in \{l, r\}$. My models contain data about the motion trajectory of both the left hand and the right hand; by allowing two sets of parameters, I allow the motion trajectory of the left hand to be scaled and shifted separately from the motion trajectory of the right hand (so, for example, ϕ^l refers to the current position in the model for the left hand's trajectory, while ϕ^r refers to the position in the model for the right hand's trajectory). In summary, there are 7 parameters that describe each state.

Initialization. The sample set is initialized with N samples distributed over possible starting states and each assigned a weight of $\frac{1}{N}$. Specifically, the initial parameters are picked uniformly according to:

$$\begin{aligned} \mu &\in [1, \mu_{max}] \\ \phi^i &= \frac{1 - \sqrt{y}}{\sqrt{y}}, y \in [0, 1] \\ \alpha^i &= [\alpha_{min}, \alpha_{max}] \\ \rho^i &\in [\rho_{min}, \rho_{max}] \end{aligned} \tag{1}$$

Prediction. In the prediction step, each parameter of a randomly sampled s_t is used to s_{t+1} determine based on the parameters of that particular s_t . Each old state, s_t , is randomly chosen from the sample set, based on the weight of each sample. That is, the weight of each sample determines the probability of its being chosen. This is done efficiently by creating a cumulative probability table, choosing a uniform random number on $[0, 1]$, and then using binary search to pull out a sample (see Isard and Blake for details [1]). The following equations are used to choose the new state :

$$\begin{aligned} \mu_{t+1} &= \mu_t \\ \phi_{t+1}^i &= \phi_t^i + \rho_t^i + N(\sigma_\phi) \\ \alpha_{t+1}^i &= \alpha_t^i + N(\sigma_\alpha) \\ \rho_{t+1} &= \rho_t^i + N(\sigma_\rho) \end{aligned} \tag{2}$$

where $N(\sigma_*)$ refers to a number chosen randomly according to the normal distribution with standard deviation σ_* . This adds an element of uncertainty to each prediction, which keeps the sample set diffuse enough to deal with noisy data. For a given drawn sample, predictions are generated until all of the parameters are within the accepted range. If, after, a set number of attempts it is still impossible to generate a valid prediction, a new sample is created according to the initialization procedure above.

Updating. After the Prediction step above, there exists a new set of N predicted samples which need to be assigned weights. The weight of each sample is a measure of its likelihood given the observed data $Z_t = (z_t, z_{t_1}, \dots)$. We define $Z_{t,i} = (z_{t,i}, z_{(t-1),i}, \dots)$ as a sequence of observations for the i th coefficient over time; specifically, let $Z_{(t,1)}, Z_{(t,2)}, Z_{(t,3)}, Z_{(t,4)}$ be the sequence of observations of the horizontal velocity of the left hand, the vertical velocity of the left hand, the horizontal velocity of the right hand, and the vertical velocity of the right hand respectively. Extending Black and Jepson [2], we then calculate the weight by the following equation:

$$p(z_t | s_t) = \prod_{i=1}^4 p(Z_{t,i} | s_t) \tag{3}$$

where $p(z_{t,i}|s_t) = \frac{1}{\sqrt{2\pi}} \exp \frac{-\sum_{j=0}^{\omega-1} (z_{(t-j),i} - \alpha^* m_{(\phi^* - \rho^* j),i}^\mu)^2}{2(\omega-1)}$ and where ω is the size of a temporal window that spans back in time. Note that ϕ^* , α^* and ρ^* refer to the appropriate parameters of the model for the blob in question and that $\alpha^* m_{(\phi^* - \rho^* j),i}^\mu$ refers to the value given to the i th coefficient of the model μ interpolated at time $\phi^* - \rho^* j$ and scaled by α^* .

3 Gesture Model and Image Preprocessing

We adopt the two gesture model to verify the proposed particle filter. As shown in Figure 2, gesture 1 means leftover and gesture 2 means paddle.

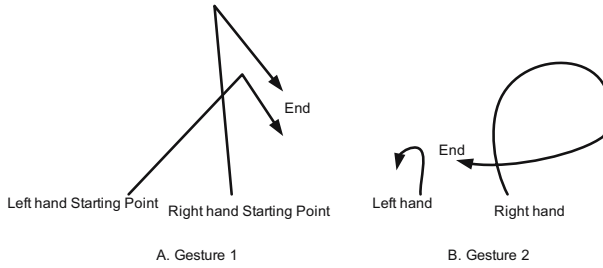


Fig. 2. Two gesture model

3.1 Raw Image Preprocessing

The image sequences were filmed using a Sony DCR Camcoder. They were manually aligned and then converted into sequences of TIFs to be processed in MATLAB. Each TIF was 320x240 pixels, 24bit color. The lighting and background in each sequence is held constant; the background is not cluttered. The focus of my project was not to solve the tracking problem, hence I wanted the hands to be relatively easy to track. I collected 7 film sequences of each sign(see Figure 3).



Fig. 3. Gesture Images of the Two Models

3.2 Skin Extraction

In order to segment out skin-colored pixels, we used the color segment routine we developed in MATLAB. Every image in every each sequence was divided into the following regions: skin, background, clothes, and outliers. First of all, we set up the mask using the gaussian distribution based on mean and covariance value which is stored in the database. Then we segment the images into four section above mentioned regions. So, we get the the segment of skin as shown in Figure 4.

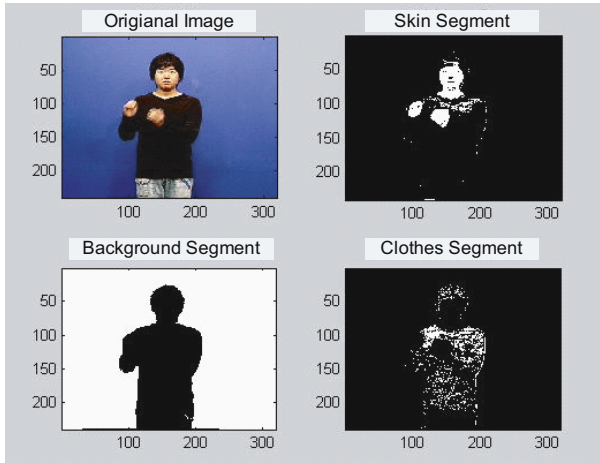


Fig. 4. Output of Segmentation

3.3 Finding Skin-Colored Blobs

We then calculated the centroid of the three largest skin colored 'blobs' in each image. Blobs were calculated by processing the skin pixel mask generated in the previous step. A blob is defined to be a connected region of 1's in the mask. Finding blobs turned out to be a bit more difficult than we had originally thought. Our first implementation was a straightforward recursive algorithm which scans the top down from left to right until it comes across a skin pixel which has yet to be assigned to a blob. It then recursively checks each of that pixel's neighbors to see if they too are skin pixels. If they are, it assigns them to the same blob and recurses. On such large images, this quickly led to stack overflow and huge inefficiency in MATLAB.

3.4 Calculating the Blobs' Motion Trajectories over Time

At this point, tracking the trajectories of the blobs over time was fairly simple. For a given video sequence, we made a list of the position of the centroid for each of the 3 largest blobs in each frame. Then, we examined the first frame in the sequence and determined which centroid was farthest to the left and which was farthest to the right. The one on the left corresponds to the right hand of signer, the one to the right corresponds to the left hand of the signer. Then, for each successive frame, we simply determined which centroid was closest to each of the previous left centroid and called this the new left centroid; we did the same for the blob on the right. Once the two blobs were labelled, we calculated the horizontal and vertical velocity of both blobs across the two frames using $[(\text{change in position})/\text{time}]$. We recorded these values for each sequential frame pair in the sequence. The example of the tracking is shown in Figure 5.

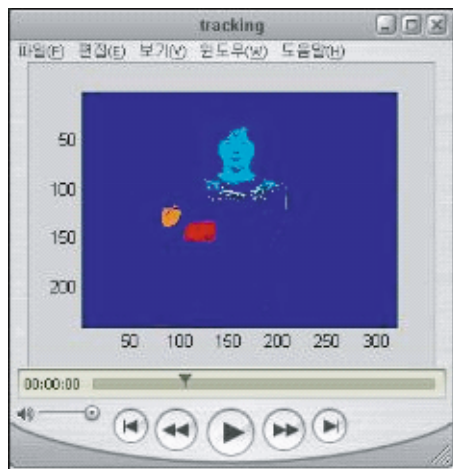


Fig. 5. Tracking result using centroid calculation

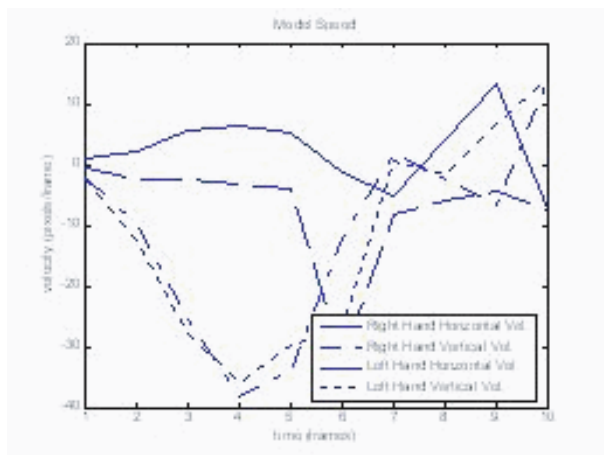


Fig. 6. Velocity of Model I

3.5 Creating the Motion Models

We then created models of the hand motions involved in each sign. Specifically, for each frame in the sign, we used 5 training instances to calculate the average horizontal and vertical velocities of both hands in that particular frame. The following graphs show the models derived for both signs (see Figure 6).

4 Experimental Results

To test the proposed particle filter scheme, we used two gesture models which are shown in Figure 2 in this paper. The coefficients of the particle filter are $\mu_{max} = 2$,

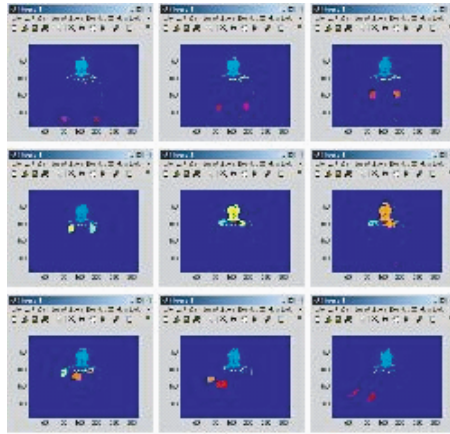


Fig. 7. The Tracking process of particle filter for the model 1(From left to right, top to down)

$\alpha_{min} = 0.5, \alpha_{max} = 1.5, \rho_{min} = 0.5, \rho_{max} = 1.5$ to maintain the 50Also, the other parameters are settled by $\sigma_\phi = \sigma_\alpha = \sigma_\rho = 0.1$. The variable of ω equation 3 is 10.

5 Conclusions

In this paper, we have developed the particle filter for the gesture recognition. This scheme is important in providing a computationally feasible alternative to classify the gesture in real time. We have proved that given an image, particle filter scheme classify the gesture in real time.

Acknowledgements. This work was supported by the Korea Research Foundation Grant funded by the Korean Government(MOEHRD) (KRF-2006-D00356).

References

1. Isard, M., Blake, A.: CONDENSATION-conditional Density Propagation for Visual Tracking. *International Journal of Computer Vision* 29(1), 5–28 (1998)
2. Black, M.J., Jepson, A.D.: A Probabilistic Framework for Matching Temporal Trajectories: Condensation-based Recognition of Gestures and Expressions. In: *Proceedings 5th European Conf. Computer Vision*, vol. 1, pp. 909–924 (1998)
3. Isard, M., Blake, A.: A Mixed-state Condensation Tracker with Automatic Model-switching. In: *Proceedings 6th International Conference of Computer Vision*, pp. 107–112 (1998)
4. Lee, Y.W.: Adaptive Data Association for Multi-target Tracking Using Relaxation. *LNCS*, vol. 3564, pp. 552–561 (2005)
5. Lee, Y.W., Seo, J.H., Lee, J.G.: A Study on the TWS Tracking Filter for Multi-Target Tracking. *J. KIEE* 41(4), 411–421 (2004)

Cooperative Approaches to Bacterial Foraging Optimization

Hanning Chen^{1,2}, Yunlong Zhu¹, Kunyuan Hu¹, Xiaoxian He³, and Ben Niu^{1,2}

¹ Key Laboratory of Industrial Informatics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, 110016, China

² School of Graduate, Chinese Academy of Sciences, Beijing, 100039, China

³ College of Information Science & Engineering, Central South University, Changsha, 410083, China

{chenhanning, ylzhu}@sia.cn, xxhe@mail.csu.edu.cn

Abstract. Bacterial Foraging Optimization (BFO) is a novel optimization algorithm based on the foraging behavior of *E. coli* bacteria. This paper presents a variation on the original BFO algorithm, called Cooperative Bacterial Foraging Optimization (CBFO), which significantly improve the convergence speed, accuracy and robustness of original BFO in solving complex optimization problems. This is achieved by applying two cooperative approaches to the BFO, namely the serial heterogeneous cooperation on the implicit space decomposition level and the serial heterogeneous cooperation on the hybrid space decomposition level. Four widely-used benchmark functions have been implemented to test the proposed algorithm, and the results show remarked improvement in performance over the original BFO.

Keywords: Bacterial Foraging Optimization; Cooperative Bacterial Foraging Optimization; Implicit space decomposition; Explicit space decomposition.

1 Introduction

In recent years, bacterial foraging behaviors (i.e. bacterial chemotaxis), as a rich source of potential engineering applications and computational model, have attracted more and more attentions. A few models have been developed to mimic the bacterial foraging behaviors and been applied for solving practical problems [1, 2, 3]. Among them, Bacterial Foraging Optimization (BFO) is a population-based numerical optimization algorithm in the literature. Currently, BFO has been applied successfully to some engineering problems, such as optimal control [4], harmonic estimation [5], transmission loss reduction [6] and machine learning [7], etc. However, experimentation with complex and multimodal benchmark functions reveal that the BFO algorithm possesses a poor convergence behavior compared to other nature-inspired algorithms and its performance also heavily decreases with the growth of the search space dimensionality.

It should be noted that even the most successful nature-inspired optimization techniques (e.g. Genetic Algorithm (GA) [8] and Particle Swarm Optimization (PSO) [9])

are also sensitive to the increase of the problem complexity and dimensionality, due to their stochastic nature. Cooperative search is one of the solutions to this problem that have been extensively studied in the past decade [12, 13]. The basic approach involves having more than one search module running and exchanging information among each other in order to explore the search space more efficiently and reach better solutions [10]. In order to improve the BFO's performance on complex optimization problems, this paper applies cooperative search technique to the BFO and presents the Cooperative Bacterial Foraging Optimization (CBFO) algorithm. In order to evaluate the performance of the CBFO, extensive studies based on a set of 4 widely-used benchmark functions have been carried out. For comparison purposes, we also implemented the original BFO, the standard PSO and the simple real-coded GA on these functions respectively. The simulation results are encouraging: the CBFO algorithm shows remarked performance improvement over the original BFO.

The rest of the paper is organized as follows. Our Cooperative Bacterial Optimization algorithm will be introduced and its implementation details will be described in Section 2. Section 3 tests the algorithms on the benchmark functions, and gives out the results. Finally, section 4 outlines the conclusions.

2 Cooperative Bacterial Foraging Optimization

This work proposes two variants of Cooperative Bacterial Foraging Algorithm, namely CBFO-S and CBFO-H. According to the taxonomies on cooperation search algorithms in [10], they can be classified into the serial heterogeneous cooperation on the implicit space decomposition level and the serial heterogeneous cooperation on the hybrid space decomposition level respectively.

2.1 The CBFO-S Algorithm

As indicate in [3], the bacterium with a large run-length unit parameter has the exploring ability, while the bacterium with a relatively small run-length unit parameter has the exploiting skill. This inspired us to divide the foraging procedure of artificial bacteria colony into multiple phases, and each phase occupies a portion of generations and characterized by the different value of run-length unit parameter C .

In CBFO-S, different BFO algorithms (with different run-length unit parameters) execute in sequential phases. The output of each BFO (the best positions found by each bacterium in each phase) supplies as an input to the algorithm in the next phase. In the initial phase, the bacteria colony search the whole solution space with a large C , which permit the bacteria to locate promising regions and avoid trapped in local optima. Each bacterium records all its visited positions in this phase and the position with the highest fitness value is considered as a promising solution candidate. When entrance into the next phase, the bacteria colony is reinitialized with relatively small C from these promising solution candidates and start exploiting the promising regions (the neighborhoods of these candidates) until the needed criteria for switching to the next phase is reached. Then a bacteria colony is reinitialized again with even smaller C to fine-tune the best-so-far solutions found in the foregoing phase. Hence, the CBFO-S algorithm can be classified into the serial heterogeneous cooperation on the implicit space decomposition level.

Table 1. Pseudocode for CBFO-S

INITIALIZE: the position and the associated run-length unit C_{initial} of the bacteria colony;

For (each phase $k = 1 : N_p$)

For (each chemotactic step $t = 1 : N_c$)

For (each bacterium $i = 1 : S$)

Calculate the fitness $J_i(t, k)$ of i^{th} bacterium;

TUMBLE: Generate a random vector Δ , where each element belongs to $[-1, 1]$.

 1]. Move the bacterium i in the direction of $\Delta / \sqrt{\Delta^T \Delta}$ by a unit walk of size $C(k)$. Then **calculate** the new fitness $J_i(t+1, k)$ of bacterium i ;

 Set $flag = 0$;

RUN: While ($flag < N_s$)

If ($J_i(t+1, k) < J_i(t, k)$)

 Take another unit walk in the same direction;

Calculate the new fitness as $J_i(t+1, k)$;

$flag = flag + 1$;

End if

End while

End for

REPRODUCTION: The $S/2$ bacteria with the worst fitness die and the other $S/2$ bacteria with the best fitness split;

ELIMINATION & DISPERSAL: With probability p_{eds} , eliminate and disperse each bacterium;

End for

REINITIALIZE: bacteria positions from the potential candidate positions found by each bacterium in the phase k .

EVOLUTION: Evolution is added to run-length unit by:

$C(k+1) = C(k)/\alpha$; // α is user-defined constant.

End for

The pseudocode of CBFO-S is described in Table 1, where N_p indicates the number of evolutionary phases, N_c represents the number of chemotactic steps in a bacterium's life time, S is the bacteria colony size and N_s is the maximum number of steps in the process of Run. We also embed the reproduction, elimination and dispersal processes into each chemotactic step. This can speed up the algorithm convergence rate significantly.

2.2 The CBFO-H Algorithm

The CBFO-H Algorithm consists of two search stages working in a serial manner. The first stage, which applied the BFO model with a large run-length unit parameter C_L , run for a number of iterations then pass the best found solutions to the next stage.

The second stage reinitializes the bacteria colony in these best-so-far positions with a smaller run-length unit parameter C_S , and applies the explicit space decomposition cooperative approach to the BFO. This approach relies on splitting the search space (n -dimensional vector) into $n/2$ subspaces (which is 2-dimensional vector), where each subspace is searched by a separate bacteria colony. The overall solution is the

Table 2. Pseudocode for CBFO-H

Stage 1: the BFO algorithm

INITIALIZE: the position and the associated run-length unit C_L of the bacteria colony;

For (each chemotactic step $t = 1 : N_c^{s1}$)

For (each bacterium $i = 1 : S$)

TUMBLE;

RUN;

REPRODUCTION;

ELIMINATION & DISPERSAL;

End for

End for

PASS the best found solutions of each bacterium to stage 2;

Stage 2: the multi-colony cooperative BFO algorithm using explicit space decomposition

REINITIALIZE: bacteria positions from the best found solutions and the associate run-length unit C_S .

SPLIT the whole population into $n/2$ separate colonies of 2-D vectors;

For (each chemotactic step $t = 1 : N_c^{s2}$)

For (each colony $j = 1 : n/2$)

For (each bacterium $i = 1 : S$)

TUMBLE;

RUN;

REPRODUCTION;

ELIMINATION & DISPERSAL;

End for

UPDATE the best bacterium replace the j^{th} component of the overall solution;

End for

EVOLUTION: Evolution is added to run-length unit by:

If ($t \bmod \beta = 0$) // β is user-defined constant.

$C(t+1) = C(t)/\alpha$; // α is user-defined constant.

End if

End for

vector combining the best bacterium of each colony. This algorithm works by sequentially traverse the colonies: to evolve all the bacteria in colony j , the other $n/2-1$ components in the overall solution are kept constant (with their values set to the global best bacteria from the other $n/2-1$ colonies); then the j^{th} colony evolves and replace the j^{th} component of the overall solution by its best bacterium.

The pseudocode of CBFO-H is described in Table 2, where N_c^{s1} and N_c^{s2} represent the number of chemotactic steps in stage 1 and 2 respectively.

3 Experiments

3.1 Benchmark Functions

The set of benchmark functions contains four functions that are commonly used in evolutionary computation literature [11] to show solution quality and convergence rate. The formulas of these functions are listed below.

1. Sphere function

$$f_1(x) = \sum_{i=1}^n x_i^2 \quad x \in [-5.12, 5.12] \quad (1)$$

2. Rosenbrock function

$$f_2(x) = \sum_{i=1}^n 100 \times (x_{i+1} - x_i^2)^2 + (1 - x_i)^2 \quad x \in [-2.048, 2.048] \quad (2)$$

3. Rastrigrin function

$$f_3(x) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i)) + 10 \quad x \in [-5.12, 5.12] \quad (3)$$

4. Griewank function

$$f_4(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad x \in [-600, 600] \quad (4)$$

3.2 Parameter Settings for the Involved Algorithms

Experiment was conducted to compare four algorithms, namely the original BFO, the proposed CBFO-H, the real-coded GA and the PSO with inertia weight on the four benchmark functions with 10 dimensions. The parameters setting for BFO, and CBFO-H are summarized in Table 3. The parameters N_{re} and N_{ed} of BFO and N_c^{s2} of CBFO-H can be adjusted to change the total chemotactic steps of each algorithm on each benchmark function.

The PSO algorithm we used is the standard one (i.e. the global version with inertia weight). The parameters were given by the default setting of [11]: the acceleration factors c_1 and c_2 were both 2.0; and a decaying inertia weight ω starting at 0.9 and ending at 0.4 was used. The population size was set at 50 for the PSO algorithm.

The GA algorithm we executed is a real-coded Genetic Algorithm with intermediate crossover and Gaussian mutation. The population of the GA is 100 and all the control parameters, e.g. mutation rate and crossover rate, etc., were set to be the same of [14].

Table 3. Parameters of the BFO algorithms

Type	BFO	CBFO-H
S	50	50
N_c	100	–
N_s	4	4
C	$10^{-3} \times R$	–
P_{ed}	0.1	0.1
α	–	10
C_L	–	$10^{-2} \times R$
C_S	–	$10^{-4} \times R$
N_c^{s1}	–	200
β	–	100

Table 4. Comparison among CBFO-H, BFO, PSO and GA on 10-D problems

10D		BFO	CBFO-H	PSO	GA
f_1	Mean	19.6331	0	0	0.0118
	Std	5.4090	0	0	0.0094
f_2	Mean	12.0991	1.4813e-007	0.7294	8.6634
	Std	9.5047	4.6694e-008	1.4964	0.5557
f_3	Mean	36.3513	0.0111	10.8450	4.9512
	Std	10.5818	0.0257	4.5956	1.1516
f_4	Mean	99.7775	0.2702	62.9737	0.3439
	Std	24.8497	0.0846	12.2581	0.1538

3.3 Results for the 10-D Problems

This experiment was conducted on 10-D problems to compare the proposed CBFO-H algorithm with the original BFO, real-coded GA and Particle Swarm Optimization. This experiment runs 30 times respectively for each algorithm on each benchmark function. The total numbers of chemotactic steps (or iterations) were set to be 1000 for Sphere ($N_{re} = 5$ and $N_{ed} = 2$ for BFO; $N_c^{s2} = 800$ for CBFO-H), 3000 for Rosenbrock

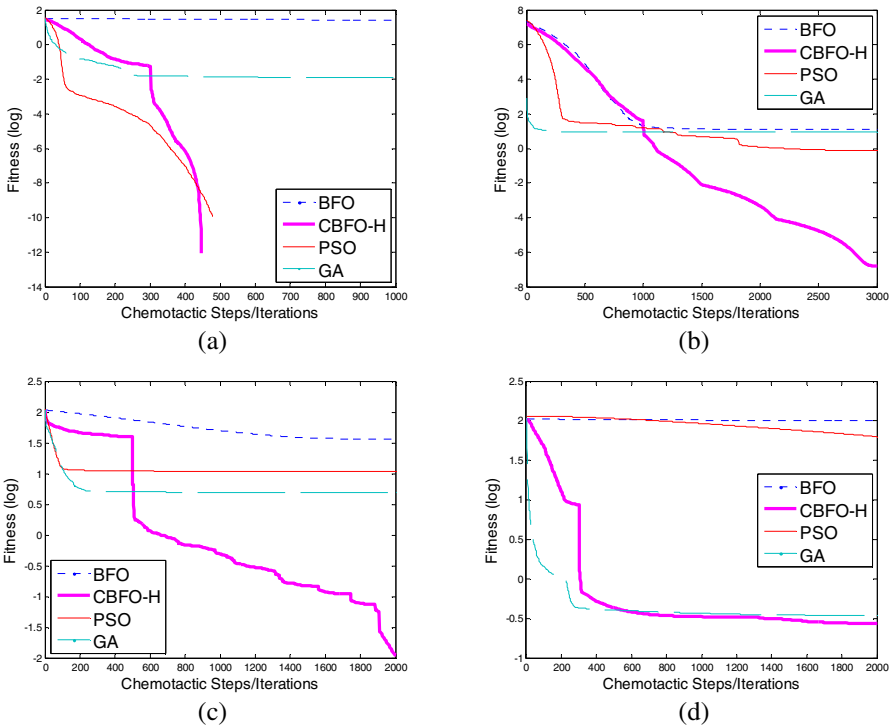


Fig. 1. Convergence results of CBFO-H, BFO, PSO and GA on 10-D benchmark functions. (a) Sphere; (b) Rosenbrock; (c) Rastrigrin; (d) Griewank.

($N_{re} = 5$ and $N_{ed} = 6$ for BFO; $N_c^{s2} = 2800$ for CBFO-H) and 2000 for Rastrigrin and Griewank ($N_{re} = 5$ and $N_{ed} = 4$ for BFO; $N_c^{s2} = 1800$ for CBFO-H). Table 4 lists the experimental results (i.e., the mean and standard deviations of the function values found in 30 runs) for each algorithm on functions $f_1 \sim f_4$. If the average optimum value is less than 10^{-10} , it is shown as 0 in the table. Fig. 4 shows the search progresses of the average values found by all algorithms over 30 runs for functions $f_1 \sim f_4$.

From the results, CBFO-H surpasses all other algorithms on functions $f_1 \sim f_4$. This significant improvement is due to the ability of BFO (with large run-length unit) to locate promising regions in stage 1, as well as the fine-grained search property of explicit space decomposition in stage 2, both of which are combined in our proposed algorithm.

4 Conclusions

This paper applied the cooperative approaches to the Bacterial Foraging Optimization (BFO) and proposed the Cooperative Bacterial Foraging Optimization (CBFO) model with two variants, namely CBFO-S and CBFO-H, which can be classified into the serial heterogeneous cooperation on the implicit space decomposition level and the serial heterogeneous cooperation on the hybrid space decomposition level respectively. These cooperative approaches used here resulted in a significant improvement in the performance of the Bacterial Foraging Optimization algorithm in terms of convergence speed, accuracy and robustness.

Four widely used benchmark functions have been used to test the CBFO-H algorithm in comparison with the original BFO, the stand PSO and the real-coded GA. The simulation results are encouraging: the CBFO-H are definitely better than the original BFO for all the test functions and appears to be comparable with the PSO and GA.

There are ways to improve our proposed algorithms. The further research efforts should focus on the tuning of the user-defined parameters for CBFO algorithms based on extensive evaluation on many benchmark functions and real-world problems. Moreover, the self-adaptive mechanisms for the parameters of CBFO may be worthy to undertake to obtain some additional improvements (e.g. which can remove the need of specifying a particular parameter setting of CBFO for a particular problem).

Acknowledgments. This work was supported by the National 863 plans projects of China under Grant 2006AA04A124 and the National 863 plans projects of China under Grant 2006AA04A117.

References

1. Bremermann, H.J., Anderson, R.W.: An Alternative to Back-propagation: A Simple Rule of Synaptic Modification for Neural Net Rraining and Nemory. Technical Report PAM-483, Center for Pure and Applied Mathematics, University of California (1990)
2. Müeller, S., Marchetto, J., Airaghi, S., Koumoutsakos, P.: Optimization Based on Bacterial Chemotaxis. IEEE Trans. on Evolutionary Computation 6(1), 16–29 (2002)

3. Passino, K.M.: Biomimicry of Bacterial Foraging for Distributed Optimization and Control. *IEEE Control System Magazine*, 52–67 (2002)
4. Kim, D.H., Cho, J.H.: Adaptive Tuning of PID Controller for Multivariable System Using Bacterial Foraging Based Optimization. In: Szczepaniak, P.S., Kacprzyk, J., Niewiadomski, A. (eds.) *AWIC 2005. LNCS (LNAI)*, vol. 3528, pp. 231–235. Springer, Heidelberg (2005)
5. Mishra, S.: A Hybrid Least Square-fuzzy Bacterial Foraging Strategy for Harmonic Estimation. *IEEE Trans. Evolutionary Computation* 9(1), 61–73 (2005)
6. Tripathy, M., Mishra, S., Lai, L.L., Zhang, Q.P.: Transmission Loss Reduction Based on FACTS and Bacteria Foraging Algorithm. *PPSN*, pp. 222–231 (2006)
7. Kim, D.H., Cho, C.H.: Bacterial Foraging Based Neural Network Fuzzy Learning. In: *II-CAI*, pp. 2030–2036 (2005)
8. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Publishing Corporation, Inc., Reading (1989)
9. Kennedy, J., Eberhart, R.C.: *Swarm Intelligence*. Morgan Kaufmann Publishers, San Francisco (2001)
10. El-Abd, M., Kamel, M.: A Taxonomy of Cooperative Search Algorithms. In: Blesa, M.J., Blum, C., Roli, A., Sampels, M. (eds.) *HM 2005. LNCS*, vol. 3636, pp. 32–41. Springer, Heidelberg (2005)
11. Shi, Y., Eberhart, R.C.: Empirical Study of Particle Swarm Optimization. In: *Proceedings of the IEEE Congress on Evolutionary Computation*, Piscataway NJ, pp. 1945–1950 (1999)
12. Potter, M., Jong, K.D.: A Cooperative Coevolutionary Approach to Function Optimization. In: *Proceedings from PPSN V*, pp. 530–539. Springer, Heidelberg (1994)
13. Bergh, F., Engelbrecht, A.P.: A Cooperative Approach to Particle Swarm Optimization. *IEEE Transactions on Evolutionary Computation* 8(3), 225–239 (2004)
14. Sumathi, S., Hamsapriya, T., Surekha, P.: *Evolutionary Intelligence: An Introduction to Theory and Applications with Matlab*. Springer, Heidelberg (2008)

Crossbar Composite Spring-Nets to Optimize Multi-Agent Systems and Computer Networks

Dianxun Shuai

East China University of Science and Technology, Shanghai 200237, China
shdx411022@online.sh.cn

Abstract. Elastic nets (EN) have been effectively used in the traveling salesman problem (TSP), the prediction problem of protein structure, and so on.¹ Nevertheless the limitations of the EN theory and its architecture seriously prevent the EN approach from the application for problem-solving of multi-agent systems (MAS) and computer networks (CN). This paper presents a crossbar composite spring-nets (CCSN) approach to MAS and CN, which transforms the optimization problem of MAS and CN into the evolutionary dynamics of crossbar composite spring nets. The CCSN approach is essentially different from EN and has many advantages over EN in terms of the problem-solving performance and the suitability for complex environment in MAS and CN.

Keywords: elastic nets; crossbar composite spring nets, artificial intelligence, multi-agent system, distributed problem solving.

1 Introduction

Basically there are two problem-solving categories for multi-agent systems (MAS) and computer networks (CN): one is concerned with how to increase the global outcome of the system through the cooperation among individual agents/entities regardless of their personal payoffs; and another is that each agent/entity tries to autonomously and rationally increase its own personal utility with no common goal, no global control strategy, and even no overall consistent information. In other words, all agents in MAS and CN are regarded either to be unselfish or selfish. Most of coordination approaches [1]-[5] to MAS and CN are based on symbolic logic, and only have lower-degree parallelism. Furthermore they only take into account the simple, bilateral, aware, and conscious interactions among agents/entities, such as cooperation and competition. But both selfish agents and unselfish agents should be considered for problem-solving in MAS and CN. Moreover, the different agents may exhibit the different degree of autonomy and rationality in the sense of the intention strength for the global benefit or individual utility. More complicated, unilateral, unaware, and even unconscious interactions among agents/entities also should be considered for MAS and CN.

¹ This work was supported by the National Natural Science Foundation of China under Grant No.60575040 and No.60473044.

The elastic nets (EN) algorithms [5]-[13] are used for global optimization problems, such as traveling salesman problem, the prediction problem of protein structure from amino acid sequences, and modeling problem of visual procession and track recognition. On the other hand, however, the current EN approaches to problem-solving have many limitations and shortcomings. The nodes in EN algorithms possess the fixed positions, and have no direct interaction among them. The EN only has a single-loop homogeneous elastic band with uniform coarseness everywhere. Moreover, the EN elongates gradually only by the attractive forces, without repelling force and other anisoelastic forces. The above mentioned limitations seriously prevent the EN algorithms from applying to multi-agent systems and computer networks.

This paper presents a crossbar composite spring-nets (CCSN) approach to MAS and CN, which is entirely different from the EN with respect to theory and architecture. But the traditional EN may be regarded as the simplest case of CCSN. The CCSN approach overcomes many limitations of EN and has many advantages over EN in terms of the problem-solving performance and the suitability for complex environment in MAS and CN.

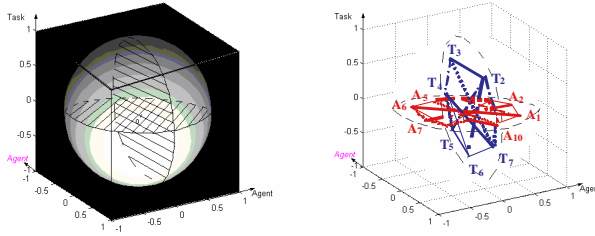
2 Crossbar Composite Spring Nets for MAS and CN

The task allocation and resource assignment in MAS and CN may be formalized by a matrix A , as shown in Table 1. There are a finite set $\mathcal{A} = \{A_1, \dots, A_n\}$ of n service agents, and a finite set $\mathcal{T} = \{T_1, \dots, T_m\}$ of m task agents, with each service agent A_i having a resource capacity vector $\mathbf{r}_i = [r_{ik}(t)]_{1 \times h}$, and each task agent T_j requiring a resource vector $\mathbf{d}_j = [d_{jk}(t)]_{1 \times h}$ and being able to afford payoff c_j . At the time t , the service agent A_i provides the task agent T_j with a resource vector $\mathbf{a}_{ij} = [a_{ijk}(t)]_{1 \times h}$, and the task agent T_j offers a payoff vector $\mathbf{p}_{ij} = [p_{ijk}(t)]_{1 \times h}$ for a unit resource vector to the service agent A_i .

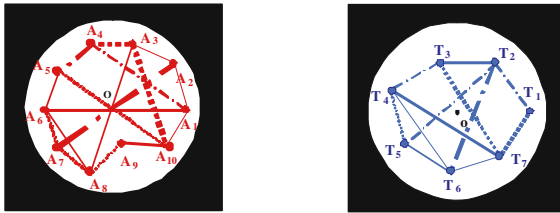
Table 1. The representation matrix A of task allocation and resource assignment in multi-agent systems and computer networks

Service Agent \ Task Agent	Task Agent				
	T_1	\dots	T_j	\dots	T_m
A_1	$\mathbf{a}_{11}, \mathbf{p}_{11}$	\dots	$\mathbf{a}_{1j}, \mathbf{p}_{1j}$	\dots	$\mathbf{a}_{1m}, \mathbf{p}_{1m}$
\vdots	\vdots		\vdots		\vdots
A_i	$\mathbf{a}_{i1}, \mathbf{p}_{i1}$	\dots	$\mathbf{a}_{ij}, \mathbf{p}_{ij}$	\dots	$\mathbf{a}_{im}, \mathbf{p}_{im}$
\vdots	\vdots		\vdots		\vdots
A_n	$\mathbf{a}_{n1}, \mathbf{p}_{n1}$	\dots	$\mathbf{a}_{nj}, \mathbf{p}_{nj}$	\dots	$\mathbf{a}_{nm}, \mathbf{p}_{nm}$

The optimal matrix A for MAS and CN should minimize an appropriately defined energy function under the following circumstances: There may be a variety of social coordinations among service agents and among task agents, including unilateral, unaware behavior; Distinct agents may have different autonomy and rationality in the context of the concern about its personal profit or the whole



(a) Crossbar composite spring net (CCSN)



(b) Horizontal composite spring net (c) Vertical composite spring net

(A circle button represents an agent node; Black square with hollow circumference represents the gravitational field; A solid line, broken line and dash line represent a compression spring, extension spring and unilateral spring, respectively. The thickness of a line represents the strength of the spring force.)

Fig. 1. The architecture of a crossbar composite spring net for task allocation and resource assignment in MAS (n=10, m=7)

utility of the system; According to the market economy regularity, a service agent and a task agent may adjust its service strategy and payoff strategy, respectively.

The physical model of crossbar composite spring nets for distributed problem solving in MAS and CN is illustrated in Fig.1. All the service agent nodes and task agent nodes lie in a horizontal composite spring net and a vertical composite spring net, respectively. Service agent nodes and task agent nodes are evenly distributed at an even radian along a horizontal circle and vertical circle, respectively. For simplicity, suppose that each node only moves along a radial orbit. Although there is no direct connection between the horizontal and vertical spring nets, but they implicitly interact with each other through the market mechanism.

Springs among nodes in the horizontal spring net or vertical spring net represent the social interactions among the corresponding service agents or task agents, respectively. The strength and force-deformation property of a spring depend upon the intensity and type of social interaction. We can make use of an appropriate composite spring to easily represent many kinds of the social

interactions based on mind-reading among agents, including the competition, cooperation, enticement, deception, avoidance, exploitation, coalition, reciproca- tion, interference, collaboration, habituation, compromise, preference, etc. The spring force asymmetrically acting on two ends of a spring can describe uni- lateral interactions, such as enticement, avoidance, exploitation, deception etc., which is different from the spring widely used in mechanics where the force of a spring must act equally on its two ends. Moreover, we can easily design a spring that has the time-varying non-linear force-deformation property to outline the more complicated social interaction among autonomous agents if necessary. In the simplest case, for example, an extension spring can represent by using its pulling force the competition between two agents; and a compression spring can describe by using its pushing force the cooperation between two agents, with the force strength reflecting the interaction intensity.

Each agent node is exerted simultaneously by the gravitational field of the circumference that surrounds the agent node, and by the forces that represent the interactions with other agent nodes in the same spring net. The radial distance between a node and the circumference represents the personal utility of the corresponding agent, i.e., the profit of a service agent or the charge of a task agent, under the current situation of MAS. It turns out that, in order to maximize the personal utility of an agent, the corresponding agent node in a composite spring net will try to move as near as possible to the circumference that surrounds the agent node.

3 Evolution of Crossbar Composite Spring Nets

For the service agent $A_i \in \mathcal{A}$ and task agent $T_j \in \mathcal{T}$, their personal energy functions, f_i and g_j , are defined respectively by

$$f_i = \exp \left[- \sum_{j=1}^m \mathbf{a}_{ij} \mathbf{p}_{ij}^T \right]; \quad g_j = 1 - \exp \left[- \sum_{i=1}^n \mathbf{a}_{ij} \mathbf{p}_{ij}^T \right], \quad (1)$$

where f_i and g_j are represented by the radial distance from the corresponding agent node to the circumference in the spring net where the agent node lies.

The potential energy functions, E_G and F_G , of the horizontal and vertical gravitational field are defined respectively by:

$$E_G = k^2 \ln \sum_{i=1, A_i \in \mathcal{A}}^n \exp[f_i^2/2k^2]; \quad F_G = k'^2 \ln \sum_{j=1, T_j \in \mathcal{T}}^m \exp [g_j^2/2k'^2], \quad (2)$$

where $0 \leq k, k' < 1$.

The aggregate energy functions, J_h and J_v , of the horizontal and vertical spring net are defined respectively by:

$$J_h = \alpha \sum_{i=1}^n f_i + \beta_1 \sum_{i=1}^n \left\| \sum_{j=1}^m \mathbf{a}_{ij} - \mathbf{r}_i \right\|^2 + \beta_2 \sum_{j=1}^m \left\| \sum_{i=1}^n \mathbf{a}_{ij} - \mathbf{d}_j \right\|^2, \quad (3)$$

$$J_v = \alpha' \sum_{j=1}^m g_j - \beta \sum_{j=1}^m \left[c_j - \sum_{i=1}^n \mathbf{a}_{ij} \mathbf{p}_{ij}^T \right], \quad (4)$$

where α, β_1, β_2 and α', β are positive constant.

The interactive potential energy functions between two service agent nodes and between two task agent nodes are denoted respectively as follows: At the time t , the interaction of A_i with respect to A_j gives rise to interactive potential energy $E_{ij}^i(f_i, f_j)$ for A_i and $E_{ij}^j(f_i, f_j)$ for A_j , respectively. Similarly, The interaction of T_i with respect to T_j yields interactive potential energy $F_{ij}^i(g_i, g_j)$ for T_i and $F_{ij}^j(g_i, g_j)$ for T_j , respectively. Generally, $E_{ij}^i(f_i, f_j)$ and $E_{ij}^j(f_i, f_j)$ may be a time-varying non-linear function of f_i and f_j . In general, $E_{ij}^i(f_i, f_j)$ and $E_{ij}^j(f_i, f_j)$ depend on the kind and strength of the interaction of A_j with respect to A_i . There may be $E_{ij}^i(f_i, f_j) \neq E_{ij}^j(f_i, f_j)$, $E_{ij}^i(f_i, f_j) \neq E_{ji}^i(f_j, f_i)$, and $E_{ij}^j(f_i, f_j) \neq E_{ji}^j(f_j, f_i)$. Particularly, the unilateral interaction of A_i with respect to A_j can give rise to $E_{ij}^i(f_i, f_j) = 0$ or $E_{ij}^j(f_i, f_j) = 0$. For example, the enticement or deception behavior of A_i with respect to A_j will cause $E_{ij}^i(f_i, f_j) = 0$ and $E_{ij}^j(f_i, f_j) > 0$; the avoidance or forbearance behavior of A_i with respect to A_j will cause $E_{ij}^i(f_i, f_j) > 0$ and $E_{ij}^j(f_i, f_j) = 0$. For competition, both will be positive; and for cooperation negative.

In summary, the agent nodes, A_i and T_j , may move along their own radial orbit under the influence of the personal energy, the aggregate energy, the potential energy, and the interactive potential energy.

Upon the intension degree for the above factors, distinct agent nodes may exhibit different degree of autonomy and personality. The agent A_i will change its resource assignment vector $\mathbf{a}_{ij}(t)$ by

$$\Delta \mathbf{a}_{ij} = -\tau_{ij} \frac{\partial f_i}{\partial \mathbf{a}_{ij}} - \lambda_{ij} \frac{\partial E_G}{\partial \mathbf{a}_{ij}} - \rho_{ij} \frac{dJ_h}{d\mathbf{a}_{ij}} - \zeta_{ij} \frac{\partial \sum_{q=1}^n E_{iq}}{\partial \mathbf{a}_{ij}} \quad (5)$$

where τ_{ij} , λ_{ij} , ρ_{ij} , ζ_{ij} are all the non-negative coefficients less than 1; and $E_{iq} = E_{iq}^i(f_i, f_q) + E_{qi}^i(f_q, f_i)$. We therefore obtain the radial velocity of service agent node A_i along its radial orbit by

$$v_i = \frac{df_i}{dt} = \sum_{j=1}^m \frac{\partial f_i}{\partial \mathbf{a}_{ij}} \frac{d\mathbf{a}_{ij}^T}{dt} = -f_i \sum_{j=1}^m \frac{d\mathbf{a}_{ij}}{dt} \mathbf{p}_{ij}^T \approx -f_i \sum_{j=1}^m \Delta \mathbf{a}_{ij} \mathbf{p}_{ij}^T. \quad (6)$$

The task agent T_j will change its payoff vector \mathbf{p}_{ij} by

$$\Delta \mathbf{p}_{ij} = -\tau'_{ij} \frac{\partial g_j}{\partial \mathbf{p}_{ij}} - \lambda'_{ij} \frac{\partial F_G}{\partial \mathbf{p}_{ij}} - \rho'_{ij} \frac{dJ_v}{d\mathbf{p}_{ij}} - \zeta'_{ij} \frac{\partial \sum_{q=1}^n F_{jq}}{\partial \mathbf{p}_{ij}} \quad (7)$$

where λ'_{ij} , τ'_{ij} , ρ'_{ij} , ζ'_{ij} are all the non-negative coefficients less than 1; and $F_{jq} = F_{jq}^j(g_j, g_q) + F_{qj}^j(g_q, g_j)$. We therefore obtain the radial velocity of task agent node T_j along its radial orbit by

$$v'_j = \frac{dg_j}{dt} = \sum_{i=1}^n \frac{\partial g_j}{\partial \mathbf{p}_{ij}} \frac{d\mathbf{p}_{ij}^T}{dt} = (1 - g_j) \sum_{i=1}^n \frac{d\mathbf{p}_{ij}}{dt} \mathbf{a}_{ij}^T \approx (1 - g_i) \sum_{i=1}^n \Delta \mathbf{p}_{ij} \mathbf{a}_{ij}^T. \quad (8)$$

In what follows we give properties related to the horizontal composite spring net and vertical composite spring net, for the page length the detailed proofs omitted.

Lemma 1. *In the horizontal spring net (resp. the vertical spring net), if k (resp. k') is very small, the decrease of the potential energy E_G (resp. F_G) of gravitational field will result in the increase of the minimal profit of service agents (resp. the decrease of the maximal payoff of task agents).*

Lemma 2. *The behavior of the service agent node A_i (resp. the task agent node T_j) that is related to the second term of Eq.(5) (resp. Eq.(7)) always brings about the increase of the minimal personal profit of service agents (resp. the decrease of the maximal personal payoff of task agents), with the intensity being directly proportional to the coefficient λ_{ij} (resp. λ'_{ij}).*

Theorem 1. *The behavior of the service agent node A_i (resp. task agent node T_j) caused by the first and second terms of Eq.(5) (resp. Eq.(7)) always results in the increase of its own personal profit (resp. the decrease of its own personal payoff), with the intensity being directly proportional to $(\lambda_{ij}\omega_i f_i + \tau_{ij})$ (resp. $(\lambda'_{ij}\omega'_i g_i + \tau'_{ij})$).*

Theorem 2. *The behavior of the service agent node A_i (resp. task agent node T_j) caused by the third term of Eq.(5) (resp. Eq.(7)) increases the global profit of service agents (resp. decrease the global charge of task agents), with the intensity being directly proportional to ρ_{ij} (resp. ρ'_{ij}).*

Theorem 3. *The behavior of service agent node A_i (resp., task agent node T_j) caused by the fourth term of Eq.(5) (resp. Eq.(7)) decreases the interactive potential energy, with the intensity being proportional to ζ_{ij} (resp. ζ'_{ij}).*

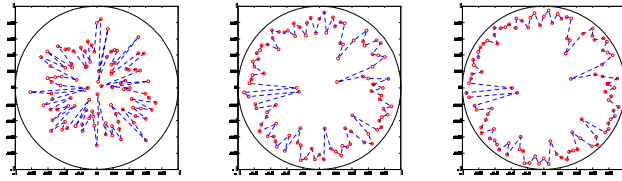
Theorem 4. *The evolutionary process of the horizontal spring net described by the Eq.(5), (6) (resp. vertical spring net, by Eq.(7), (8)) can realize the distributed problem solving in MAS in the context that distinct service agent (resp. task agent) may have different autonomy and personality with respect to the personal profit, aggregate profit (resp. personal payoff, aggregate payoff) and social interactions.*

4 Simulations and Conclusions

During the process of problem solving, the evolutionary trajectories of service agent nodes in the horizontal composite spring net and task agent nodes in the vertical composite spring net are illustrated in Fig.2. The influence of autonomy and personality (selfishness) of agents on the performance of MAS is shown in Fig.3.

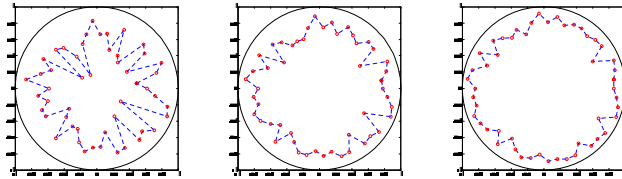
We conclude as follows. The crossbar composite spring nets approach to distributed problem-solving in MAS and CN is essentially different from the conventional elastic net. The proposed CCSN approach is featured as follows:

- very high parallelism and real-time computational performance;
- An agent or entity in MAS and CN is regarded neither as being fully selfish nor as being fully unselfish. In this sense, distinct agents can exhibit quite different autonomous degrees.
- A variety of complicated social interactions among agents can be taken into account in the process of problem-solving.



$t_1 = 1$ $t_1 = 75$ $t_1 = 150$

(a) The trajectories of service agent nodes in the horizontal spring net.
(the number of service agent nodes :100)



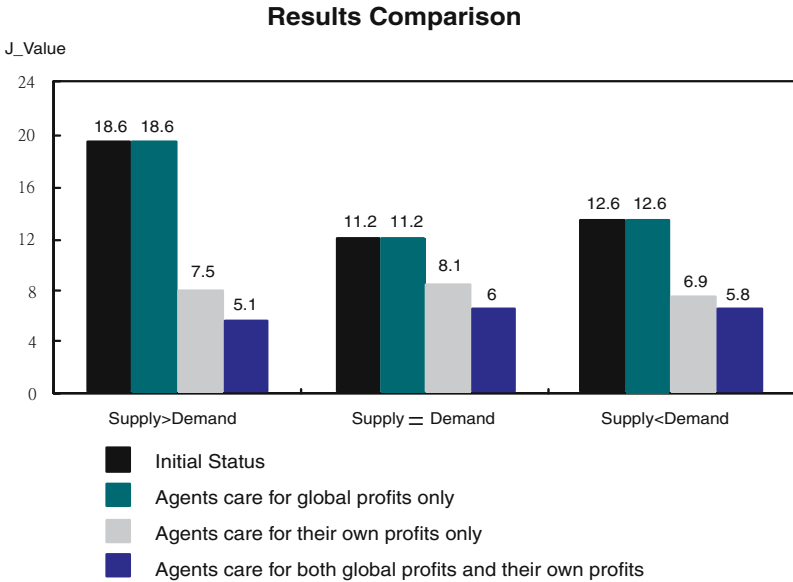
$t_1 = 1$ $t_1 = 24$ $t_1 = 75$

(b) The trajectories of task agent nodes in the vertical spring net.
(the number of task agent nodes :50)

Problem Class: the whole resources possibly supplied by service agents are more than whole demands practically required by task agents. The number of service agents : 100; The number of task agents : 50. Parameters: $k = 0.8, k' = 0.8, \alpha = 1, \beta_1 = 0.1, \beta_2 = 0.1, \alpha' = 1, \beta = 0.1, Z = 150, z_h = 0.001, z_v = 0.001$; and $\lambda_{ij}, \lambda'_{ij}, \tau_{ij}, \tau'_{ij}, \rho_{ij}, \rho'_{ij}, \zeta_{ij}, \zeta'_{ij}, u_{ij}, u'_{ij}, v_{ij}, v'_{ij}$ are at random within $[0,1]$.

Fig. 2. The trajectories of nodes in crossbar composite spring nets

- The social interactions among agents can take place randomly, emergently and concurrently.
 - The microscopic characterization of an individual agent can be combined with the macroscopic property of MAS and CN.
- The proposed CCSN approach also overcomes many limitations of the traditional elastic net in the following senses:
- The agent nodes can move at a certain velocity under the resultant forces;
 - There exist the various complicated social interactions between agents.
 - The composite springs have much more flexibility than the uniform isotropic band of the traditional elastic net.
 - The spring net elongates gradually under various kinds of forces, rather than only under elastic forces.
 - The conventional elastic net can be regarded as a simplest case of the crossbar composite spring net, which has the fixed agents (i.e., the cities for TSP), uniform width of elastic band, and single loop of net form.
 - The time-varying property of an agent can be considered.



$J_{value} = J_h + J_v$. Parameters: $k = 0.8, k' = 0.8, \alpha = 1, \beta_1 = 0.1, \beta_2 = 0.1, \alpha' = 1, \beta = 0.1, Z = 200, z_h = 0.1, z_v = 0.1$; and $\lambda_{ij}, \lambda'_{ij}, \tau_{ij}, \tau'_{ij}, \rho_{ij}, \rho'_{ij}, \zeta_{ij}, \zeta'_{ij}, u_{ij}, u'_{ij}, v_{ij}, v'_{ij}$ are at random within $[0,1]$; The number of problems: more than 100; This figure shows the results of 3 typical problems;

Fig. 3. The performance comparisons with respect to different selfish degree of agents, by using the crossbar composite spring net for the task allocation and resource assignment in MAS

References

1. Chaudhury, A.: Two Mechanisms for Distributed Problem Solving. *IEEE Trans. on System, Man, and cybernetics* 28(1), 48–55 (1998)
2. Kraus, S., Wilkenfeld, J., Zlotkin, G.: Multiagent Negotiation under Time Constraints. *Artificial Intelligence* 75, 295–345 (1995)
3. Shehory, O., Kraus, S.: Methods for Task Allocation via Agent Coalition Formation. *Artificial Intelligence* 101, 165–200 (1998)
4. Shehory, O., Kraus, S.: A Kernel-Oriented Model for Coalition-Formation in General Environments: Implementation and Results. In: *Proc. AAAI 1996, Portland*, pp. 134–140 (1996)
5. Durbin, R., Willshaw, D.: An Analogue Approach to The Traveling Salesman Problem Using an Elastic Net Method. *Nature* 326, 91–689 (1987)
6. Willshaw, D.J., Von der, M.C.: A Marker Introduction Mechanism for The Establishment of Ordered Neural Mapping: Its Application to The Retinotchal Problem. *Philosophical Transactions of the Royal Society of London, Series B* 287, 43–203 (1977)
7. Burr, D.J.: An Improved Elastic Net Method for the Traveling Salesman Problem. In: *IEEE International Conference on Neural Networks*, vol. 1, pp. 69–76. IEEE Press, New York (1988)

8. Stone, J.V.: The Optimal Elastic Net: Finding Solutions for The Traveling Salesman. *Artificial Neural Networks* 2, 1077–1080 (1992)
9. Keith, D.B., Burak, E., Ken, A.D.: The Elastic Net Algorithm and Protein Structure Prediction. *Journal of Computational Chemistry* 23(1), 77–83 (2002)
10. Loe, K.F.: Topographical Mapping Forms of Objects into Gaussian Elastic Net. *Neurocomputing* 11, 171–178 (1996)
11. Kisel, I., Kovalenko, V.: Elastic Net for Broken Multiple Scattered Tracks. *Computer Physics Communication* 98, 45–51 (1996)
12. Kisel, I., Kovalenko, V., Laplanche, F.: Cellular Automaton and Elastic Net for Event Reconstruction in the NEMO-2 Experiment. *Nuclear Instruments and Methods in Physics Research (Section A)* 387, 433–442 (1997)
13. Kisel, I., Konotopskaya, E., Kovalenko, V.: Elastic Neural Net for Track and Vertex Search. *Nuclear Instruments and Methods in Physics Research (Section A)* 389, 167–168 (1997)
14. Bui, D.L., Greenshaw, T.J., Schmidt, G.: A Combination of an Elastic Net and a Hopfield Net to Solve the Segment Linking Problem in the Forward Tracker of the H1 Detector at HERA. *Nuclear Instruments and Methods in Physics Research (Section A)* 389, 184–186 (1997)

Formation Control of Multiple Robots Using Parametric and Implicit Representations

Yesim H. Esin, Mustafa Unel, and Mehmet Yildiz

Faculty of Engineering and Natural Sciences, Sabanci University
34956 Istanbul, Turkey
yesime@su.sabanciuniv.edu,
munel@sabanciuniv.edu,
meyildiz@sabanciuniv.edu

Abstract. A novel method is presented for formation control of a group of autonomous mobile robots using parametric and implicit descriptions of the desired formation. Shape formation is controlled by using potential fields generated from Implicit Polynomial (IP) representations and the control for keeping the desired shape is designed using Elliptical Fourier Descriptors (EFD). Coordination of the robots is modeled by linear springs between each robot and its nearest two neighbors. This approach offers more flexibility in the formation shape and scales well to different swarm sizes and to heterogeneous systems. The method is simulated on robot groups with different sizes to form various formation shapes.

Keywords: Shape Formation; Robot Coordination; Autonomous Mobile Robots.

1 Introduction

The role of the autonomous robots in our lives is increasing in many fields. There are many applications that a group of multiple robots can achieve much better than a single robot. There has been significant research on groups of autonomous mobile robots which can be used for applications including tasks such as exploration [1], surveillance [2], search and rescue [3], mapping of unknown or partially known environments [4], distributed manipulation [5], and transportation of large objects [6]. An important issue in these tasks is the flexibility in the definition of the working area of robots.

This paper introduces a shape formation method which is highly flexible for various tasks. The robots are initially positioned randomly in a defined area. The aim is to make the robots to form and keep a desired shape. During this formation, the robots are also desired to coordinate among themselves to avoid collisions and keep the desired distance between each other.

Our proposed method is based on a parametric description of the desired formation by using elliptic Fourier descriptors (EFD) and its implicit polynomial (IP) representation [7]. Implicit polynomial description of the curve is used to

define an algebraic distance between each robot and the desired curve in the first phase in which robots are attracted to form the desired shape. The parametric description is used for trajectory tracking in the second phase in which the robots are controlled to keep the formation. Besides these controls, a coordination control to keep the desired distances between the neighboring robots is also introduced. In this work, each mobile robot is modeled as a point particle.

The main advantage of this approach over other approaches is its flexibility on the desired curve. The desired curve can be any reasonable closed curve. Some other methods are rigid in formation constraints [8] [9]. Another advantage of our method is that it offers more flexibility on the number of robots in the group and the heterogeneousness of the swarms compared to other methods such as [10] [11].

The presented method is simulated in Matlab. The simulation program is written to be modular so that the simulations can be done with any desired number of robots and with any desired closed curve. The performance of the proposed method has been verified by simulations with different sizes of the swarms.

Organization of this paper is as follows: Representation of the free-form curves is presented in Section 2. Designing a control for shape formation, keeping the formation and coordination of robots are presented in Section 3. Section 4 presents the simulation results and discussions. Conclusions are presented in Section 5.

2 Parametric and Implicit Representation of Formation

It is known that closed-bounded curves can be represented by elliptic Fourier descriptors (EFD) [12]. The elliptic Fourier description of a closed curve is given as follows:

$$x(t) = a_o + \sum_{k=1}^n (a_k \cos kt + b_k \sin kt) . \quad (1)$$

$$y(t) = c_o + \sum_{k=1}^n (c_k \cos kt + d_k \sin kt) . \quad (2)$$

In this equation, n is a positive number which represents the number of the harmonics used to represent the closed curve. As it is well-known, the efficiency of the representation increases with increasing number of harmonics. On the other hand, the computational cost increases with the number of the harmonics. Therefore, an optimal decision on the number of the harmonics should be made when modeling a complex curve by considering both the efficiency in the description and the computational cost.

The desired formation shape can be represented both by a parametric function using elliptic Fourier descriptors and by an implicit function, $F(x, y) = 0$, obtained through the implicitization of EFDs using the method detailed in [7]. The implicit function is good for producing potential functions to be used for formation shaping and the parametric function is employed for keeping the formation.

The representation of implicit curve has the form $F(x, y) = \sum_{0 \leq i+j \leq d} a_{ij} x^i y^j = 0$ where a_{ij} are the coefficients and d is the degree of the polynomial respectively. As detailed in [7], the resulting implicit curve has degree $d = 2n$ where n is the number of harmonics used in the parametric representation of the curve.

3 Control of Mobile Robots

In this study, the robots are modeled as point particles. The kinematic model for the i^{th} robot is as follows:

$$\begin{pmatrix} \dot{x}_i \\ \dot{y}_i \end{pmatrix} = \begin{pmatrix} u_{total}^i \\ v_{total}^i \end{pmatrix}. \quad (3)$$

where \dot{x}_i and \dot{y}_i are the velocities of the particle in the x and y directions respectively with respect to the world coordinate frame. In this control, $\begin{pmatrix} u_{total}^i \\ v_{total}^i \end{pmatrix}$ is obtained by the sum of the control components designed for the formation control and coordination control, namely

$$\begin{pmatrix} u_{total}^i \\ v_{total}^i \end{pmatrix} = \begin{pmatrix} u_{formation}^i \\ v_{formation}^i \end{pmatrix} + \begin{pmatrix} u_{coord}^i \\ v_{coord}^i \end{pmatrix}. \quad (4)$$

where $\begin{pmatrix} u_{formation}^i \\ v_{formation}^i \end{pmatrix}$ is formation component of the control. The design of this formation component changes with respect to the phase of the formation, if the robots are in the phase to form the desired shape or to keep the formation. $\begin{pmatrix} u_{coord}^i \\ v_{coord}^i \end{pmatrix}$ is the coordination component of the control.

3.1 Formation Control Using Implicit Polynomial Potential Functions

Initially, robots are randomly positioned in the defined area and the aim is to form the desired formation shape. For the design of the formation component of control input, the implicit polynomial representation of the curve is used. The position error function between the i^{th} robot and the curve is given by the algebraic distance to the curve using the implicit equation as

$$e_{form}^i = F(x_i, y_i). \quad (5)$$

where e_{form}^i is the position error with respect to the desired curve defined for i^{th} robot and x_i and y_i are the x and y position of the robot with respect to the world coordinate frame. The control will be designed to force the error to decrease exponentially as

$$\dot{e}_{form}^i = -\lambda e_{form}^i. \quad (6)$$

where λ is a positive number. Substituting $e_{formation}^i = F(x_i, y_i)$ into the equation above yields

$$\dot{F}(x_i, y_i) = -\lambda F(x_i, y_i) . \tag{7}$$

Using chain rule of differentiation, we have

$$F_x(x_i, y_i)\dot{x}_i + F_y(x_i, y_i)\dot{y}_i = -\lambda F(x_i, y_i) . \tag{8}$$

which can be rewritten as

$$(F_x(x_i, y_i) \ F_y(x_i, y_i)) \begin{pmatrix} \dot{x}_i \\ \dot{y}_i \end{pmatrix} = -\lambda F(x_i, y_i) . \tag{9}$$

Taking the kinematic model of the robots into account, the optimal formation control can be determined using pseudo inverse, namely

$$\begin{pmatrix} u_{formation}^i \\ v_{formation}^i \end{pmatrix} = -\lambda \frac{1}{\|\nabla F(x_i, y_i)\|^2} F(x_i, y_i) \begin{pmatrix} F_x(x_i, y_i) \\ F_y(x_i, y_i) \end{pmatrix} . \tag{10}$$

3.2 Keeping Formation Using Elliptic Fourier Descriptors

After the robots reach the desired formation, a new control is designed to keep the robots in the desired formation. In the design of this control component, the parametric representation of the desired formation is used. The error function for this phase is defined as

$$e_{formation}^i = \begin{pmatrix} x_i^*(t) \\ y_i^*(t) \end{pmatrix} - \begin{pmatrix} x_i(t) \\ y_i(t) \end{pmatrix} . \tag{11}$$

$$\Rightarrow \dot{e}_{formation}^i = \begin{pmatrix} \dot{x}_i^*(t) \\ \dot{y}_i^*(t) \end{pmatrix} - \begin{pmatrix} \dot{x}_i(t) \\ \dot{y}_i(t) \end{pmatrix} = \begin{pmatrix} \dot{x}_i^*(t) \\ \dot{y}_i^*(t) \end{pmatrix} - \begin{pmatrix} u_{formation}^i \\ v_{formation}^i \end{pmatrix} . \tag{12}$$

where $x_i^*(t)$ and $y_i^*(t)$ are the desired positions and $x_i(t)$ and $y_i(t)$ are the actual positions.


The control is designed to make the error decrease exponentially, namely:

$$\dot{e}_{formation}^i = -\lambda e_{formation}^i . \tag{13}$$

which yields

$$\begin{pmatrix} u_{formation}^i \\ v_{formation}^i \end{pmatrix} = \lambda e_{formation}^i + \begin{pmatrix} \dot{x}_i^*(t_i) \\ \dot{y}_i^*(t_i) \end{pmatrix} . \tag{14}$$

3.3 Coordination Control

The robots are appointed to be in coordination besides the shape formation task. The aim of this coordination is to avoid collisions and keep a desired distance between each robot and its neighbors. The control input of a robot for this coordination is modeled as the sum of forces of linear springs between this robot and its two nearest neighbors as in Figure 

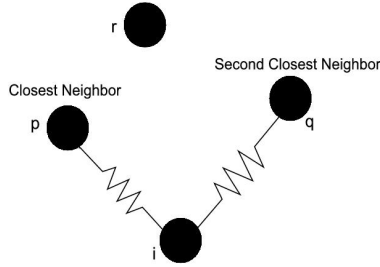


Fig. 1. Modeling of coordination control

The spring is proposed to have a normal length which is equal to the desired distance. The spring produces a force linearly proportional to the difference between the actual distance and the desired distance between the robots. The force of the spring is on the direction of the vector from the i^{th} robot to its neighbor as below:

$$\begin{pmatrix} u_{coord}^i \\ v_{coord}^i \end{pmatrix} = k(d_{desired} - d_{actual}^{ip}) \begin{pmatrix} x_i - x_p \\ y_i - y_p \end{pmatrix} + k(d_{desired} - d_{actual}^{iq}) \begin{pmatrix} x_i - x_q \\ y_i - y_q \end{pmatrix}. \tag{15}$$

where k is an adaptable spring constant; the spring constant is different when the robot is in shape formation or keeping formation phases. p and q are the indices for the robots that are the nearest two neighbors of i^{th} robot. d_{actual}^{ip} and d_{actual}^{iq} are the actual distances of the robot i from the robots p and q respectively. (x_p, y_p) and (x_q, y_q) are the x and y position coordinates of the robots p and q with respect to the world coordinate frame.

4 Simulation Results and Discussions

The simulations are performed in Matlab. The program is written to be modular so that the simulations can be carried out with any desired number of robots. Two different simulations will be presented. First one is a simulation of a single robot to see the efficiency of the proposed formation control. The second one is done with 5 and 6 robots on two different desired patterns to see the success of the control in both formation and coordination.

In simulations, parameters are chosen to be $\lambda = 3$, $k_{ShapeFormation} = 2$, $k_{KeepFormation} = 1$ and $d_{desired} = 0.1$.

4.1 Simulation Results for a Single Robot

In the first simulation, the proposed method is applied on a single robot. Desired pattern is represented by a Fourier descriptor function using 7 harmonics and an implicit polynomial with a degree of 14 using the methods of Section 2. The initial position of the robot is: $x = 2.5$, $y = -2$. The route of the robot can be seen in Figure 2.

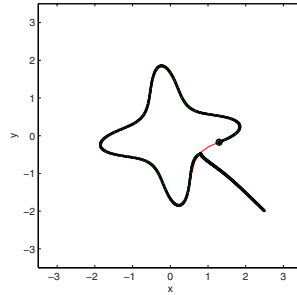


Fig. 2. Route of mobile robot

It is clearly seen that the method is successful to enable the robot to reach the desired formation curve and moves on it.

4.2 Simulation Results for 5 and 6 Robots

In this simulation, the proposed method is applied on a group of 5 and 6 robots, respectively. The desired pattern for 5 robots is an ellipse generated by a Fourier descriptor with 1 harmonic and a corresponding implicit polynomial of degree 2. Ellipse formation is shown in Figure 3. The desired pattern for 6 robots is a more complicated star shape represented by 7 harmonics and a corresponding implicit polynomial of degree 14. Formation results are depicted in Figure 4. Examination of these figures reveals the fact that although the initial positions of the robots are far away from the desired formation curve, proposed method

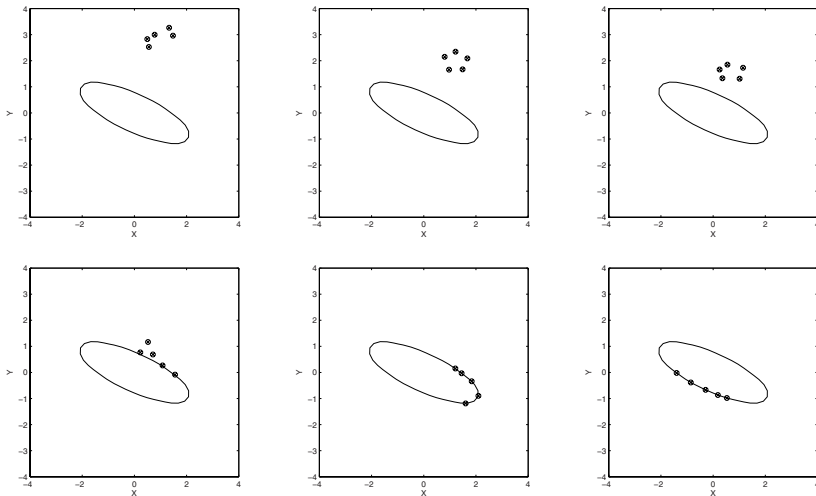


Fig. 3. Desired formation (ellipse) with 5 robots

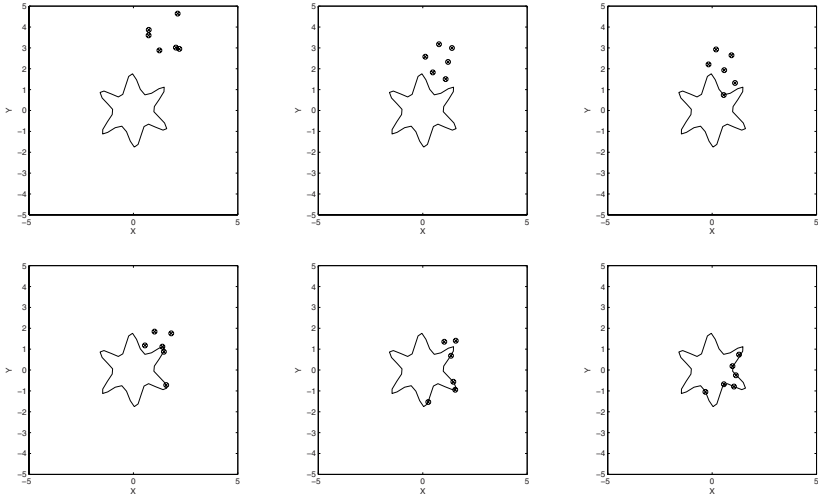


Fig. 4. Desired formation (star shape) with 6 robots

enables robots to achieve and maintain the desired formation while keeping good coordination with each other.

5 Conclusions

We have now presented a novel method for a group of mobile robots to form arbitrary desired shapes using parametric and implicit descriptions of the formation. Proposed method introduces flexibility on the desired shape and it can scale well to different swarm sizes.

We are planning to work on possible extensions of this work with non-holonomic finite size robots and implement them on actual robots.

References

1. Fox, D., Burgard, W., Kruppa, H., Thrun, S.: A Probabilistic Approach to Collaborative Multi-Robot Localization. *Auton. Robots* 8(3), 325–344 (2000)
2. Feddema, J., Schoenwald, D.: Decentralized Control of Cooperative Robotic Vehicles. *Proceedings of SPIE* 4364, 852–864 (2001)
3. Hayashi, K., Yokokohji, Y., Yoshikawa, T., Moreno, W.: Tele-existence Vision System with Image Stabilization for Rescue Robots. In: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain (2005)
4. Birk, A., Carpin, S.: Merging Occupancy Grid Maps From Multiple Robots. *Proceedings of the IEEE* 94(7), 1384–1397 (2006)
5. Manewam, T., Rittipravat, P.: Sorting Objects by Multiple Robots Using Voronoi Separation and Fuzzy Control. In: *Proceedings of the 2003 IEEE WRSJ Intl. Conference on Intelligent Robots and Systems*, Las Vegas, Nevada (2003)

6. Sugar, T., Kumar, V.: Control and Coordination of Multiple Mobile Robots in Manipulation and Material Handling Tasks. In: Corke, P., Trevelyan, J. (eds.) *Experimental Robotics VI: Lecture Notes in Control and Information Sciences*, vol. 250, pp. 15–24. Springer, New York (2000)
7. Yalcin, H., Unel, M., Wolovich, W.: Implicitization of Parametric Curves by Matrix Annihilation. *International Journal of Computer Vision* 54(1/2/3), 105–115 (2003)
8. Kang, W., Xi, N., Zhao, Y., Tan, J., Wang, Y.: Formation Control of Multiple Autonomous Vehicles: Theory and Experimentation. In: *Proceeding IFAC 15th Triennial World Congress*, pp. 1155–1160 (2002)
9. Barnes, L., Alvis, W., Fields, M., Valavanis, K., Moreno, W.: Heterogeneous Swarm Formation Control Using Bivariate Normal Functions to Generate Potential Fields. In: *Proceedings of the IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications* (2006)
10. Balch, T., Hybinette, M.: Social Potentials for Scalable Multi-Robot Formations. In: *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 85–94 (2000)
11. Leonard, N., Fiorelli, E.: Leaders, Artificial Potentials and Coordinated Control of Groups. In: *Conference on Decision and Control, Florida*, pp. 2968–2973 (2001)
12. Kuhl, F.P., Giardina, C.R.: Elliptic Fourier Features of a Closed Contour. *Computer Graphics and Image Processing* 18, 236–258 (1982)

Fuzzy SVM Training Based on the Improved Particle Swarm Optimization

Ying Li, Bendu Bai, and Yanning Zhang

School of Computer Science, Northwestern Polytechnical University, Xi'an, 710072, China
lybyp@nwpu.edu.cn

Abstract. In this paper, an improved particle swarm optimization algorithm is proposed to train the fuzzy support vector machine (FSVM) for pattern multi-classification. In the improved algorithm, the particles studies not only from itself and the best one but also from the mean value of some other particles. In addition, adaptive mutation was introduced to reduce the rate of premature convergence. The experimental results on MNIST character recognition show that the improved algorithm is feasible and effective for FSVM training.

1 Introduction

Support Vector Machines (SVMs) [1,2,3] have played a key role in broad classes of problems arising in various fields. In the conventional SVMs for pattern classification, an n -class problem is converted into n two-class problems. However, there exist unclassifiable regions in this case. To resolve this problem, Inoue *et al.* proposed the fuzzy support vector machine (FSVM) [4], in which the truncated polyhedral pyramidal membership functions was introduced to resolve unclassifiable regions.

Training a SVM requires solving a quadratic (QP) problem in a number of coefficients equal to the number of training examples. For a very large datasets, the standard numeric techniques for QP become infeasible. The practical techniques decompose the problem into the manageable subproblems over part of the data or, in the limit, perform iterative pairwise or component-wise optimization. A disadvantage of these techniques is that they may give an approximate solution, and may require many passes through the dataset to reach a reasonable level of convergence.

Particle Swarm Optimization (PSO) [5][6] is a swarm intelligent optimization method proposed by Kennedy and Eberhart. PSO introduces stochastic and multipoint searching and can find solution quickly in high dimension space. Using PSO to train SVM has been put forward recently, and this is really a novel tentative. Paquet *et al* presented the linear particle swarm optimization (LPSO) for linearly constrained problem [7], and applied this new algorithm for training SVM [8]. But the basic LPSO is not effective enough, especially in the aspect of premature convergence.

In this paper, a new improved LPSO named ILPSO is introduced for FSVM training. In ILPSO, a new particle's learning method is proposed and adaptive mutation is considered to break away from the local optimum. Experimental results on MNIST character recognition indicate the efficiency of the algorithm.

2 Fuzzy Support Vector Machines (FSVM)

2.1 Two-Class SVM

Traditionally, a SVM is a learning machine for two class classification problems, and learns from a set of example vectors x_i , and their associated classes y_i , i.e.

$$\{(x_1, y_1), \dots, (x_n, y_n)\} \subset R^N \times \{\pm 1\} \tag{1}$$

The algorithm aims to find support vectors and their corresponding coefficients to construct an optimal separating surface by the use of kernel functions in high dimension feature space. However, the nature of construct the optimal surface is to resolve a constrained quadratic programming problem for an optimal decision boundary function:

$$f(x) = \text{sgn} \left(\sum_{i=1}^n \alpha_i y_i K(x_i, x) + b \right) \tag{2}$$

where $K(\cdot, \cdot)$ is the kernel function and n is the number of the training set. The value b is a threshold. The coefficients α is the optimal solution of the quadratic programming problem:

$$\begin{aligned} \text{Max}(Q(\alpha)) &= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j K(x_i, x_j) \\ \text{st. } \sum_{i=1}^n \alpha_i y_i &= 0 \quad (i = 1, 2, \dots, n) \\ \text{st. } 0 \leq \alpha_i &\leq C \quad (i = 1, 2, \dots, n) \end{aligned} \tag{3}$$

where C is a punishment factor, which controls the balance of training error rate and the complexity of the model. It is easy to prove that the solution of this optimal problem only has part of nonzero α_i , which are corresponding to support vectors. Only support vectors affect the result of classifications.

2.2 Fuzzy SVM

In the conventional SVMs for pattern classification, an n -class problem is converted into n two-class problems. For the i th two-class problem we determine the optimal decision function $f_i(\mathbf{x})$ so that class i is separated from the remaining classes. In classification, a datum \mathbf{x} is classified into class i only when $f_i(\mathbf{x}) > 0$. In this architecture, the datum is unclassifiable if the values of more than two decision functions are positive or all the values are negative, as shown in Fig. 1. To overcome this problem, Noue *et al.* proposed the fuzzy SVM (FSVM) [4], in which one-dimensional membership function $m_{ij}(\mathbf{x})$ on the directions orthogonal to the optimal separating $f_i(x) = 0$ is defined as follows.

1. For $i = j$

$$m_{ii}(\mathbf{x}) = \begin{cases} 1 & f_i(\mathbf{x}) > 1 \\ f_i(\mathbf{x}) & \text{otherwise} \end{cases} \quad (4)$$

2. For $i \neq j$

$$m_{ij}(\mathbf{x}) = \begin{cases} 1 & f_i(\mathbf{x}) < -1 \\ -f_i(\mathbf{x}) & \text{otherwise} \end{cases} \quad (5)$$

The shape of the membership function is shown to be a truncated polyhedral pyramid^[4].

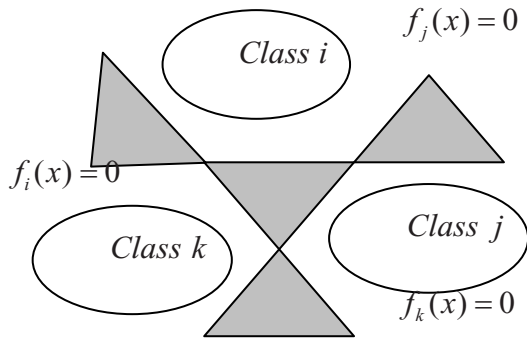


Fig. 1. Unclassifiable region by the two-class formulation

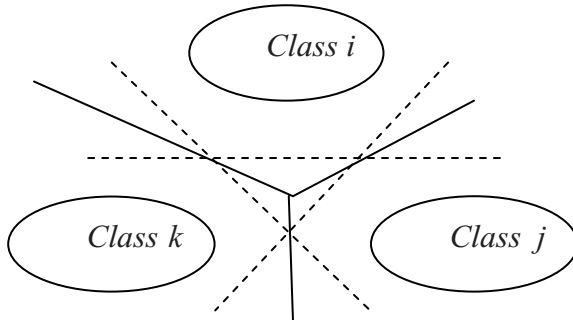


Fig. 2. Class boundary with membership functions

Then the membership function of \mathbf{x} using the minimum operator for $m_{ij}(\mathbf{x})$ ($j=1, \dots, n$)

$$m_i(\mathbf{x}) = \min_{j=1, \dots, n} m_{ij}(\mathbf{x}) \quad (6)$$

Now an unknown datum \mathbf{x} is classified into the class

$$\arg \max_{i=1, \dots, n} m_i(\mathbf{x}) \tag{7}$$

Thus, the unclassified region shown in Fig. 1 is resolved as shown in Fig. 2.

3 LPSO for Linear Equation Constrained Problem

3.1 Basic PSO

The basic PSO is as follows. The candidate solution of the problem is called a particle. Each particle has a corresponding position and velocity, denoted as $P_i = (P_{i1}, P_{i2}, \dots, P_{in})$ and $v_i = (v_{i1}, v_{i2}, \dots, v_{in})$ ($i = 1, 2, \dots, m$) respectively. where m is the number of particles and n is the dimension of the solution. Each particle also has a fitness value decided by decision function. PSO uses (8) and (9) to update the particle's velocity and position during evolution.

$$v_{i,j}^{(t+1)} = w^t * v_{i,j}^{(t)} + c_1 r_{1j}^{(t)} (pbest_{i,j}^{(t)} - p_{i,j}^{(t)}) + c_1 r_{2j}^{(t)} (pg_j^{(t)} - p_{i,j}^{(t)}) \tag{8}$$

$$p_{i,j}^{(t+1)} = p_{i,j}^{(t)} + v_{i,j}^{(t+1)} \tag{9}$$

where t is the current generation, w is the inertia weight, c_1 and c_2 are coefficients, where $c_1 = c_2 = 2.05$, and r_{1j}, r_{2j} are evenly distributed random number in $[0, 1]$, which embodies the stochastic nature of the algorithm. $pbest_i = (pbest_{i1}, pbest_{i2}, \dots, pbest_{in})$ is the best solution found by particle i whereas $pg = (pg_1, pg_2, \dots, pg_n)$ is the best found by the whole population up to now. A PSO algorithm with constriction factor^[9] has been put forward, which uses the following formula to replace above-mentioned formula (8).

$$v_{i,j}^{(t+1)} = K[v_{i,j}^{(t)} + c_1 r_{1j}^{(t)} (pbest_{i,j}^{(t)} - p_{i,j}^{(t)}) + c_1 r_{2j}^{(t)} (pg_j^{(t)} - p_{i,j}^{(t)})] \tag{10}$$

where K is called constriction factor. It has been concluded that using constriction factor while limiting the maximum velocity to the dynamic range of the position variable on each dimension is a better approach. So the PSO with constriction factor is adopted in this paper.

3.2 LPSO for Linear Equation Constrained Problem

Since the basic PSO can't be used to solve the optimization with constrains, a new LPSO algorithm for solving optimization problem with linear equation constraint is presented in [7]. In LPSO, particles update their velocity using the under-mentioned formula.

$$v_i^{(t+1)} = w^t \cdot v_i^{(t)} + c_1 r_1^{(t)} (pbest_i^{(t)} - p_i^{(t)}) + c_1 r_2^{(t)} (pg^{(t)} - p_i^{(t)}) \tag{11}$$

where the same random number r_1 and r_2 are adopted to update each component of the velocity vector, the particle's velocity is set to be zero firstly and the initial position must satisfy the linear equation constrains $AP = b, A \in R^{l \times n}, b \in R^l$.

The LPSO algorithm is summarized below.

1) Initialize. Initialize the particles such that the velocity v_i of each particle meets $v_i = 0$ and the position P_i meets $AP_i = b$.

2) Evaluate. Calculate fitness value of all particles firstly. And then compare the personal best of each particle to its current value, and set $pbest_i$ to the better one. Set the global best pg to the position of the particle with the best $pbest_i$ within the swarm.

3) Update. Update the velocity and position of each particle according to equation (11) and (9).

4) Check. While maximum iterations or minimum error rate is attained, stop the evolutions.

4 ILPSO for FSVM Training

4.1 ILPSO

Although PSO finds good solutions much faster than any other evolutionary algorithms, it usually cannot improve the quality of solutions as the number of iterations increases and it suffers from premature convergence when strongly multi-modal problems are being optimized. One of the main reasons is that all particles converge to a single point as the speed of the particles is decreased with time, thus forcing them to converge to the global best point found so far, which is not guaranteed even to be a local minimum^[10]. So we present the following improvements for LPSO to avoid the premature convergence of the swarm.

1) A new learning method is introduced. The particles not only study from itself and the best one but also from the mean value of some other particles at the beginning with bigger probability, and tend to the best one of the whole population in the later for the sake of convergence property.

2) A new adaptive mutation is introduced. Calculate the variance of the population's fitness σ^2 firstly^[11].

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^n \left(\frac{f_i - f_{avg}}{f} \right)^2 \quad (12)$$

where f_i is the fitness value of particle i , f_{avg} is the average fitness value of the population currently. f is unitary factor that controls the range of σ^2 , where $f = \max\{\max\{|f_i - f_{avg}|, 1\}, 1\}$.

And then calculate mutation probability p_m [11]:

$$p_m = \begin{cases} k, & \sigma^2 < \sigma_d^2 \\ 0, & \text{others} \end{cases} \tag{13}$$

where k is a number in $[0.1, 0.3]$, and σ_d^2 can be determined according to different actual problems.

If the probability of mutation is big enough, the position of some particles will be initialed newly.

4.2 ILPSO for Two-Class SVM Training

From the mathematical model of SVM, we can learn that the variables need to be optimized are the coefficients of support vectors, so the particle can be coded as $P = (\alpha_1, \alpha_2, \dots, \alpha_n)$ and the fitness function is selected as

$$f(P) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j K(x_i, x_j) \tag{14}$$

The steps using ILPSO to train two-class SVM are as follows:

1) Initializing all of the particles. According to the LPSO algorithm, the velocity of the particles is set to zero; the position is generated randomly with the range of $[0, C]$,

and then adjusted to the box $\sum_{i=1}^n \alpha_i y_i = 0$. Set the iteration number to zero.

2) For each particle, executing the following operations:

i. Calculating the fitness value of each particle and updating the $pbest_i$.

ii. Generating a random number r and calculating $R_t = t / Max_Gen$. If $r > R_t$,

then using

$$v_i^{(t+1)} = K[v_i^{(t)} + c_1 r_1^{(t)} (pbest_i^{(t)} - p_i^{(t)}) + c_1 r_2^{(t)} (pavg_i^{(t)} - p_i^{(t)})] \tag{15}$$

to update the velocity of the particle; otherwise, formula (12)

$$v_i^{(t+1)} = K[v_i^{(t)} + c_1 r_1^{(t)} (pbest_i^{(t)} - p_i^{(t)}) + c_1 r_2^{(t)} (pg^{(t)} - p_i^{(t)})] \tag{16}$$

is adopted. Above-mentioned $pavg_i$ is the mean value of 1/5 particles produced randomly. For any component $v_{ij} < velocity_min$ or $v_{ij} > velocity_max$, letting it follow into $[velocity_min, velocity_max]$ newly.

iii. Updating the particle's position with formula (9).

For any component $p_{ij} < 0$ or $p_{ij} > C$, letting it follow into $[0, C]$ newly, and then using the same method as in initialization to tune up the other coefficients to satisfy the linear equation constraint.

iv. Computing the value of pg .

3) Calculating the fitness of the population according to the formula (12) and the probability of mutation according to formula (13).

4) Generating a random number r . If $r < p_m^{[10]}$, choosing 1/5 particles (barring the best one) for mutation—initial the particle's position newly. Otherwise, go to step 5).

5) The current iteration number is set to $t + 1$. If $t > Max_Gen$, then pg is the solution of the problem; otherwise, return to step 2) and repeat the above procedure again.

4.3 ILPSO for FSVM Training

Based on the above ILPSO for two-class SVM, we present the following FSVM training algorithm (ILPSO_FSVM) for pattern multi-classification.

1. Apply ILPSO to train i th ($i = 1, 2, \dots, n$) two-class SVM classifier, and obtain the optimal separating function $f_i(\mathbf{x})$.
2. For \mathbf{x} , if $f_i(\mathbf{x}) > 0$ is satisfied for only one class, the input is classified into the class. Otherwise, go to step 2.
3. If $f_i(\mathbf{x}) > 0$ is satisfied for more than one class i , classify the datum into the class with the maximum $f_i(\mathbf{x})$. Otherwise, go to step 3.
4. If $f_i(\mathbf{x}) \leq 0$ is satisfied for all the classed, classify the datum into the class with the minimum absolute value of $f_i(\mathbf{x})$.

5 Experimental Results

In this section, the FSVM training algorithm based on ILPSO (ILPSO_FSVM) was tested on the MNIST dataset. The MNIST dataset is a database of optical characters, and consists of a 10-class training set of 60,000 handwritten digits. Each digit is a 28 by 28 pixel gray-level image, equivalent to a 784-dimensional input vector. Each pixel corresponds to an integer value in the range of 0 (white) and 255 (black).

Our method ILPSO_FSVM is compared with QP_FSVM, LPSO_FSVM and ILPSO_FSVM. The kernel functions of SVM in the experiment are all selected as

Table 1. Statistic testing results by different methods

Methods	Recognition number	Recognition rate
QP-FSVM	/	/
LPSO-FSVM	best mean	9390 9365.5 93.9% 93.7%
ILPSO-FSVM	best mean	9418 9387.3 94.2% 93.9%

Gaussian kernel function $K(X) = \exp(-\frac{|X - X_i|}{2\sigma^2})$. In QP_FSVM, the parameters are selected as $C = 100, \sigma = 0.5$. In LPSO_FSVM and ILPSO_FSVM, the parameters are selected as $C = 2.5, \sigma = 0.5$. In addition, the number of particles is 50. Other parameters are selected as $c_1 = c_2 = 2.05$, $K = 0.73$, $k = 0.3$, $\sigma_d^2 = 0.2$, velocity_min=-4, velocity_max=4.

Table 1 shows the experimental results of the MNIST database recognition by QP_FSVM, and by LPSO_FSVM and ILPSO_FSVM over 20 runs respectively, where “/” sign denotes that the system runs out of memory which causes the computation impossible. Obviously, ILPSO_FSVM has higher recognition rate than LPSO_FSVM.

6 Conclusion

This paper presents an improved LPSO algorithm for FSVM training. In the new algorithm, the adaptive mutation mechanism is introduced and the particle studies not only from itself and the best one but also from the mean value of some other particles with certain probability. The experiments on MNIST dataset classification show that the FSVM trained by the new algorithm can reach high recognizable rate.

Acknowledgments. This work is supported by the Specialized Research Found for the Doctoral Program of Higher Education of China (No. 20070699013), the Natural Science Foundation of Shaanxi Province of China (No. 2006F05), and the Aeronautical Science Foundation of China (No. 05I53076).

References

1. Vapnik, V.N.: The Nature of Statistical Learning Theory. Springer, New York (1995)
2. Kecman, V.: Learning and Soft Computing, Support Vector machines. Neural Networks and Fuzzy Logic Models. The MIT Press, Cambridge (2001)
3. Wang, L. (ed.): Support Vector Machines: Theory and Application. Springer, Berlin (2005)
4. Inoue, T., Abe, S.: Fuzzy Support Vector Machines for Pattern Classification. In: Proc of International Joint Conference on Neural Networks, pp. 1449–1454 (2001)
5. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: Proc. IEEE International Conference on Neural Networks, pp. 1942–1948 (1995)
6. Shi, Y., Eberhart, R.C.: A Modified Particle Swarms Optimizer. In: Proc. IEEE Congress on Evolutionary Computation, pp. 69–73 (1998)
7. Paquet, U., Engelbrecht, A.P.: A New Particle Swarm Optimizer for Linearly Constrained Optimization. In: Proc. IEEE Congress on Evolutionary Computation, pp. 227–233 (2003)
8. Paquet, U., Engelbrecht, A.P.: Training Support Vector Machines with Particle Swarms. In: Proc. International Joint Conference on Neural Networks, pp. 1593–1598 (2003)

9. Eberhart, R.C., Shi, Y.: Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization. In: Proc. IEEE Congress on Evolutionary Computation, pp. 84–88 (2000)
10. Pasupuleti, S., Battiti, R.: The Gregarious Particle Swarm Optimizer (G-PSO). In: Proc GECCO 2006, July 8–12 (2006)
11. Lu, Z.S., Hou, Z.R.: Particle Swarm Optimization with Adaptive Mutation. Chinese Journal of Electronics, 416–420 (2004)

Handling Multiobjective Problems with a Novel Interactive Multi-Swarm PSO

Yujia Wang and Yupu Yang

Department of Automation, Shanghai Jiao Tong University, Shanghai 200240,
P.R. China

Abstract. In this paper, an interactive multi-swarm PSO is proposed to handle the multiobjective optimization problems (MOPs). The preference relation of the objectives is specified during the optimization process. The selective pressure in the main swarm is improved by the preference information from the assistant swarm. The simulation results indicate that the proposed algorithm attains better performance of convergence and diversity.

1 Introduction

Particle swarm optimization (PSO) is a swarm intelligent technique inspired by the birds flock [1]. PSO has been found very efficient in solving the MOPs, and several multiobjective particle swarm optimization (MOPSO) algorithms have been proposed recently [2]. However, these techniques assume that all objectives are quantifiable, well-defined, and their relative importance remains constant. In other words, the preference of user is not considered in these techniques.

Interactive optimization is based on the idea of using human knowledge and intuition during the optimization process [3]. The expert knowledge is brought together with computational techniques by means of interactive optimization. PSO is suitable for interactive optimization because the gradient information is not used in the search process [4].

In this paper, an interactive multi-swarm PSO is proposed to handle MOPs, in which the prior preference of the user is considered. The selective pressure in the main swarm is improved by the preference information. A novel equation of updating velocity is introduced, in which the preference information from the assistant swarm is added. Thus, the preference of user is reflected in the search process.

The remainder of this paper is organized as follows. Section 2 describes the structure of PSO and gives the concept of multiobjective optimization. The details of the proposed algorithm are introduced in Section 3. In Section 4, five test functions are used to testify the proposed algorithm, and experimental results are also discussed. Conclusion is drawn in Section 5.

2 PSO and Multiobjective Optimization

2.1 Particle Swarm Optimization

PSO is based on a simplified social model [1,5] that is closely tied to swarm theory. Each particle has a position defined as $\mathbf{x} = (x_1, x_2, \dots, x_D)$ and a velocity

defined as $\mathbf{v} = (v_1, v_2, \dots, v_D)$ in the variable space. In generation $t + 1$, the velocity and position of the particle are updated as follows:

$$v_{j,t+1} = \omega v_{j,t} + c_1 r_1 (x_{pbest} - x_{j,t}) + c_2 r_2 (x_{gbest} - x_{j,t}) \quad (1)$$

$$x_{j,t+1} = x_{j,t} + v_{j,t+1} \quad (2)$$

where $j = 1, \dots, D$ is the dimension of particle; ω is the inertia weight, which influences the balance between global and local exploration abilities of the particle [6]; c_1 and c_2 are acceleration coefficients; r_1 and r_2 are random values in the range $[0, 1]$; x_{pbest} is the personal best of particle in the generation t ; x_{gbest} is the global best of population in the generation t . The performance of each particle is measured according to a pre-defined fitness function which is related to the problem concerned.

2.2 Multiobjective Optimization and the Notion of Dominance

Assuming simultaneously minimize m objectives, a multiobjective optimization problem can be described as follows:

$$\min \mathbf{y} = \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x}))$$

where $\mathbf{x} = (x_1, \dots, x_n) \in \mathbf{X}$ is an n -dimensional decision variable vector, and \mathbf{X} is the decision variable space; and $\mathbf{y} = (y_1, \dots, y_m) \in \mathbf{Y}$ where \mathbf{Y} is the objective space. Each objective depends on the decision vector \mathbf{x} .

A decision vector $\mathbf{u} \in \mathbf{X}$ is said to strictly dominate another decision vector $\mathbf{v} \in \mathbf{X}$, denoted by $\mathbf{u} \prec \mathbf{v}$, if and only if (iff) $\forall i \in \{1, \dots, m\} : f_i(\mathbf{u}) \leq f_i(\mathbf{v})$ and $\exists j \in \{1, \dots, m\} : f_j(\mathbf{u}) < f_j(\mathbf{v})$.

A decision vector $\mathbf{x} \in \mathbf{X}$ is said to be Pareto optimal with respect to \mathbf{X} iff there is no other decision vector that dominates \mathbf{x} in \mathbf{X} . The set of all Pareto optimal solutions in the decision variable space is called as Pareto optimal set. The corresponding set of objective vector is called as Pareto optimal front.

3 An Interactive Multi-swarm PSO

3.1 Classifying Objectives and the Preference Weight Value

In the design of a multiobjective problem, only a limited amount of well distributed solutions are needed by the user according to the preference. However, the majority of MOPSO algorithms [2] assume that all objectives are equally important without considering the user's preference. To reflect the preference of user, a method to classify the objectives is introduced firstly, which is based on the Pareto principle but where the relative importance of objectives can be specified according to the preference of user.

Preference relation [7, 8] can be used to estimate of the relative importance of objectives in an MOP. For every two objectives, the user is asked to specify a relation as shown in Table 1. Given a preference relation, two objectives can be evaluated using the following valuation function v :

- if $a \ll b$ then $v(a) = \alpha$ and $v(b) = \beta$;
- if $a \prec b$ then $v(a) = \gamma$ and $v(b) = \delta$;
- if $a \approx b$ then $v(a) = v(b) = \varepsilon$.

Considering the meaning of the relations, it can further be assumed that $\alpha < \gamma < \varepsilon = 1/2 < \delta < \beta$. It is also assumed that $\alpha + \beta = \gamma + \delta = 1$.

Table 1. Relation and Expression

relation	\approx	\prec	\ll
intended meaning	is equally important	is less important	is much less important

Assuming optimization m objectives, the preference weight value for each objective can be computed when the preference relation is given according to the user’s need. Firstly, for the objectives with equivalence relation, construct the equivalence classes $\{C_i | 1 \leq i \leq k\}$ and choose one element g_i from each class C_i to build up set $\mathcal{G} = \{g_1, \dots, g_k\}$, where $k \leq m$. Secondly, identify a relation matrix R with size $k \times k$. The elements of R are set as follows:

- if $g_i \ll g_j$, then $R(i, j) = \alpha$, $R(j, i) = \beta$;
- if $g_i \prec g_j$, then $R(i, j) = \gamma$, $R(j, i) = \delta$;
- if $g_i \approx g_j$, then $R(j, j) = \varepsilon$, $R(j, i) = \varepsilon$.

Finally, for each $g_i \in \mathcal{G}$, compute weight value as follows:

$$w(g_i) = \frac{S_L(g_i, R)}{\sum_{g_j \in \mathcal{G}} S_L(g_j, R)} \tag{3}$$

where $S_L(\cdot)$ is the leaving score [7] which can be computed as follows:

$$S_L(g_i, R) = \sum_{j \neq i} R(i, j). \tag{4}$$

Thus the preference weight value for each objective can be attained according to the preference of user.

3.2 Main Swarm and Assistant Swarm

A structure of multi-swarm as shown in Fig. 1 is used to handle the MOPs. The population is composed of a main swarm and several assistant swarms. Each objective is optimized by a corresponding assistant swarm and all objectives are optimized simultaneously in the main swarm. The information from assistant swarms is added to the main swarm, which indicates the preference of user.

In order to achieve “information selection”, the prior preference of user is considered to deal with the objective functions. The details of information selection are presented as follows:

- Identify the relation matrix R according to the preference of user;
- Compute the leaving score $S_L(\cdot)$ according to the relation matrix R ;

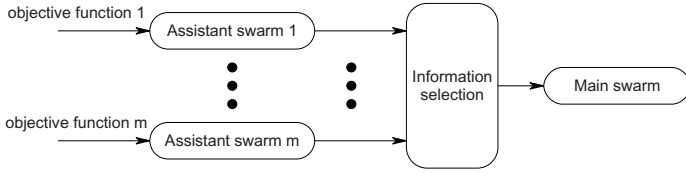


Fig. 1. The structure of multi-swarm PSO

- Use $S_L(\cdot)$ to compute the preference weight value according to equation (3) for each objective;
- Assign the preference weight values to the assistant swarms;
- Use the roulette wheel selection to identify which assistant swarm’s information is added to the main swarm.

To transfer the information from an assistant swarm to the main swarm, the velocity updating equation for each particle in the main swarm is changed as follows:

$$v_{j,t+1} = \omega v_{j,t} + c_1 r_1 (x_{pbest} - x_{j,t}) + c_2 r_2 (x_{gbest} - x_{j,t}) + c_3 r_3 (x_{agbest} - x_{j,t}) \quad (5)$$

where c_3 is an acceleration coefficient like c_1 and c_2 ; r_3 is a random value in the range $[0, 1]$; x_{agbest} is the best solution found so far by the selected assistant swarm.

The transferred direction of information is multidirectional in the subpopulation because there is interaction between all subpopulations [9,10]. But in the proposed method, the information is only transferred from assistant swarm to main swarm and the direction is unidirectional as shown in Fig. 1.

3.3 Optimization Process

The external archive is used to store the historical nondominated solutions found by the main swarm in this study. The iteration will be terminated if the number of iterations is equal to the maximum number of iterations. Thus, the optimization process can be described as follows:

1. Initialize all multi-swarms including the main swarm and assistant swarms;
2. Initialize the external archive;
3. Assign the preference weight value to each assistant swarm according to the preference of user as shown in section 3.2;
4. Evaluate each particle in all swarms;
5. Identify nondominated solutions in the main swarm and store the positions of nondominated solutions in the external archive;
6. Identify x_{gbest} from the external archive;
7. Identify the global best solution in each assistant swarm;
8. Identify x_{agbest} by roulette wheel selection according to the preference weight values of assistant swarms;
9. Update the velocities and positions of particles in the main swarm according to equation (5) and (2);

10. Update the personal best positions of particles in the main swarm according to the Pareto dominance;
11. Update the velocities and positions of particles in the assistant swarms according to equation (11) and (2).
12. If the number of iterations is less than the maximum number of iterations, return to (4).

4 Experimental Results and Discussions

4.1 Test Functions and Performance Metrics

In order to demonstrate the effectiveness of the proposed algorithm, five standard test functions are used in this study. Four of these test functions (ZDT1, ZDT2, ZDT3 and ZDT6) have two objectives, while the other one test function is DTLZ7 which has four objectives. These functions are representative of the benchmarks commonly used in the multiobjective optimization literature.

The performance of the proposed algorithm is evaluated by two functionally metrics. One is the generational distance (GD) [11], the other is the spacing (SP) [12]. GD evaluates the convergence of the nondominated solutions generated by the algorithm to the true Pareto front, and its value $GD = 0$ indicates that all solutions are on the true Pareto front. SP evaluates the range variance of neighboring solutions on the known Pareto front, and its value $SP = 0$ suggests that all solutions on the Pareto front currently available are equidistantly spaced.

4.2 Parameters Selection

Acceleration coefficient values of c_1 , c_2 and c_3 are set to be 1.5, 2.5 and 0.7 respectively in the proposed algorithm. The inertia weight ω is adjusted dynamically throughout the optimization [13], and its value is reduced as

$$\omega_{t+1} = \omega_t f_\omega \tag{6}$$

where f_ω is a constant between 0 and 1. Smaller f_ω value would result in a more dramatic reduction in ω , which would in turn result in a more local search. In this paper, $f_\omega = 0.975$ is used, which makes the algorithm have a suitably global search characteristic. The ω value is initialized to 1.4. The population size, the external archive size, the number of iterations and the number of runs are summarized in Table 2.

In order to compute the relation matrix and leaving score $S_L(\cdot)$, suppose that $\alpha = 0.05$, $\beta = 0.95$, $\gamma = 0.35$, $\delta = 0.65$, and $\varepsilon = 0.5$. The relative importance

Table 2. Population size, external archive size, number of iterations and number of runs for each test function in this study

Population size for main swarm	80	Population size for each assistant swarm	80
External archive size	80	Number of iterations	200
Number of runs	300		

of the objectives is specified. For test functions with two objectives, suppose that the first objective is less important than the second objective, i.e., $y_1 \prec y_2$. For test functions with four objectives, suppose that the relative important is $y_2 \ll y_1, y_3 \prec y_1, y_1 \prec y_4$ and $y_2 \ll y_3$.

4.3 Experimental Results and Discussions

The proposed algorithm is compared with MOPSO [14] and MOPSO-CD [15]. For MOPSO and MOPSO-CD, the parameters are set as suggested in [14] and [15], respectively. The results are reported in terms of the mean and standard deviation of the performance measures.

Table 3. Mean and standard deviation of values for GD

Algorithm	ZDT1	ZDT2	ZDT3	ZDT6	DTLZ7	ZDT1	ZDT2	ZDT3	ZDT6	DTLZ7
	mean					standard deviation				
Our Algorithm	8.4E-05	1.1E-07	8.2E-06	1.1E-07	1.2E-02	2.6E-04	4.6E-07	2.2E-05	4.2E-07	3.2E-04
MOPSO	2.5E-02	4.0E-03	7.3E-03	6.9E-03	1.8E-02	2.3E-03	6.0E-03	4.8E-04	1.0E-02	8.4E-02
MOPSO-CD	1.0E-02	1.1E-02	1.3E-02	2.8E-02	1.9E-02	3.4E-03	4.9E-05	1.5E-04	1.5E-03	8.5E-04

It can be clearly seen from Table 3, that the values of GD indicate that the proposed algorithm has obtained the best convergence for all test functions compared with MOPSO and MOPSO-CD. For DTLZ7, its Pareto front is the intersection of a straight line and a hyperplane, so it is difficult to converge to the Pareto front. However, it is worth noting that the value GD of DTLZ7 generated by the proposed algorithm is improved when the number of objectives is four. The selective pressure of particles in the main swarm is increased while x_{agbest} is added to the optimization process, thus the exploration of the proposed algorithm is improved.

Table 4. Mean and standard deviation of value for SP

Algorithm	ZDT1	ZDT2	ZDT3	ZDT6	DTLZ7	ZDT1	ZDT2	ZDT3	ZDT6	DTLZ7
	mean					standard deviation				
Our Algorithm	3.2E-03	3.8E-04	4.2E-03	8.8E-04	1.3E-02	7.3E-03	3.4E-04	1.4E-03	5.7E-04	1.9E-02
MOPSO	1.1E-02	1.0E-02	2.3E-02	2.4E-03	8.4E-02	6.8E-03	8.4E-03	4.8E-04	9.5E-04	2.8E-02
MOPSO-CD	1.6E-02	1.0E-02	1.6E-02	2.8E-02	4.8E-02	3.3E-03	3.4E-03	4.3E-03	5.7E-04	6.3E-02

Table 4 presents the mean and standard deviation values of SP for three MOPSO algorithms on five test functions. The SP values indicate that the solutions generated by the proposed algorithm obtain better distribution than the other two algorithms for all test functions. The Pareto front of ZDT3 is a nonuniform distribution, so it can be used to investigate the ability of an algorithm to maintain a good distribution. From the result of ZDT3 in Table 4, it can be seen that the performance of distribution is improved by the proposed algorithm. In order to demonstrate the distribution of the solutions on the final Pareto front, the nondominated fronts generated by the three algorithms are shown in Fig. 2 for ZDT1, ZDT2, ZDT3 and ZDT6.

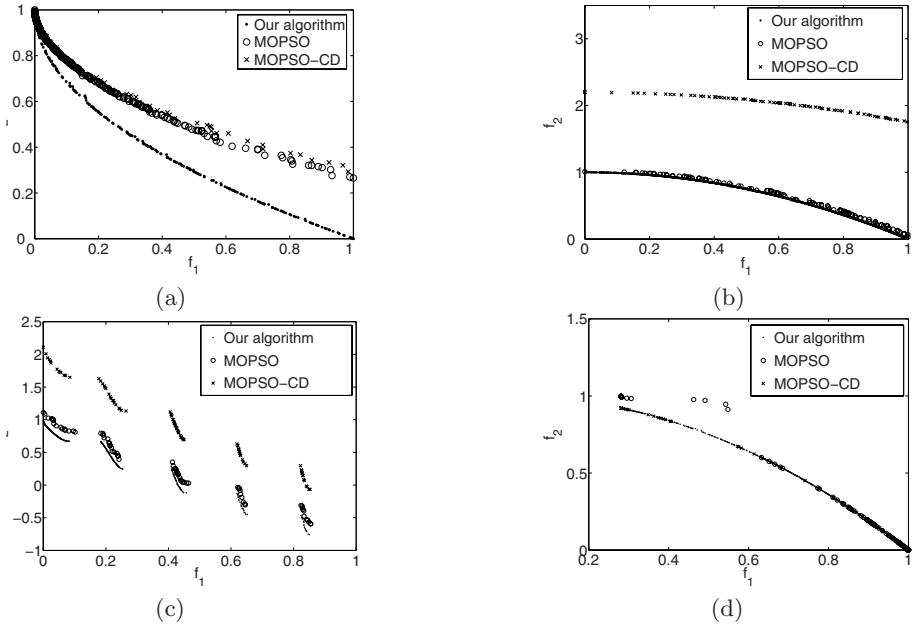


Fig. 2. Final Pareto fronts generated by the proposed algorithm, MOPSO and MOPSO-CD on test functions (a)ZDT1, (b)ZDT2, (c)ZDT3 and (d)ZDT6

5 Conclusion

An interactive multi-swarm PSO is presented to handle the multiobjective optimization problems in this paper, which considers the preference of decision maker and uses the multi-swarm as the assistant swarms to help search optimal solutions. The convergence and diversity are improved when the information from assistant swarm is added to the main swarm. The velocity updating equation is changed, which makes the algorithm attain a good balance between the exploration and exploitation. The interactive multi-swarm PSO is compared with the other two MOPSO algorithms. The results confirm that the proposed algorithm attains better convergence to the Pareto fronts for all test functions. At the same time, it is also more effective in maintaining the diversity of solutions.

References

1. Kennedy, J., Eberhart, R.: Particle Swarm Optimization. In: Proceedings of IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948 (1995)
2. Reyes Sierra, M., Coello, C.: Multi-Objective Particle Swarm Optimizers: A Survey of the State-of-the-Art. International Journal of Computational Intelligence Research 2(3), 287–308 (2006)

3. Parmee, I., Cvetkovic, D., Bonham, C., Packham, I.: Introducing Prototype Interactive Evolutionary Systems for Ill-defined, Multi-objective Design Environments. *Advances in Engineering Software* 32(6), 429–441 (2001)
4. Takagi, H., et al.: Interactive Evolutionary Computation: Fusion of the Capabilities of EC Optimization and Human Evaluation. *Proceedings of the IEEE* 89, 1275–1296 (2001)
5. Eberhart, R., Kennedy, J.: A New Optimizer Using Particle Swarm Theory. In: *Proceedings of the Sixth International Symposium on Micro Machine and Human Science, MHS*, pp. 39–43 (1995)
6. Shi, Y., Eberhart, R.: Parameter Selection in Particle Swarm Optimization. *Proceedings of Evolutionary Programming*, 591–600 (1998)
7. Parmee, I., Cvetkovic, D., Watson, A., Bonham, C.: Multiobjective Satisfaction within an Interactive Evolutionary Design Environment. *Evolutionary Computation* 8(2), 197–222 (2000)
8. Chen, S., Hwang, C., Beckmann, M., Krelle, W.: *Fuzzy Multiple Attribute Decision Making: Methods and Applications*. Springer, New York (1992)
9. Lovbjerg, M., Rasmussen, T., Krink, T.: Hybrid Particle Swarm Optimiser with Breeding and Subpopulations. In: *Proceedings of the third Genetic and Evolutionary Computation Conference, GECCO 2001*, vol. 1, pp. 469–476 (2001)
10. van den Bergh, F., Engelbrecht, A.: A Cooperative Approach to Particle Swarm Optimization. *IEEE Transactions on Evolutionary Computation* 8(3), 225–239 (2004)
11. Deb, K., Jain, S.: Running Performance Metrics for Evolutionary Multi-objective optimization. Technical report, KanGAL Report (2002)
12. Schott, J.: Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization. Master's thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge (1995)
13. Venter, G., Sobieszczanski-Sobieski, J.: Multidisciplinary Optimization of a Transport Aircraft Wing Using Particle Swarm Optimization. *Structural and Multidisciplinary Optimization* 26(1), 121–131 (2004)
14. Coello, C., Pulido, G., Lechuga, M.: Handling Multiple Objectives with Particle Swarm Optimization. *IEEE Transactions on Evolutionary Computation* 8(3), 256–279 (2004)
15. Raquel, C., Naval Jr., P.: An Effective Use of Crowding Distance in Multiobjective Particle Swarm Optimization. In: *Proceedings of the conference on Genetic and evolutionary computation*, pp. 257–264. ACM Press, New York (2005)

The Explicit Exploration Information Exchange Mechanism for Niche Technique*

Jun Zhang^{1,2} and Kwok-Wing Chau³

¹ Institute of Intelligent Machines, Chinese Academy of Sciences, P.O.Box 1130, Hefei, Anhui, 230031, China

² School of Electronic Science and Technology, Anhui University, Hefei Anhui, China

³ Department of Civil & Structural Engineering, The Hong Kong Polytechnic University, Hungghom, Kowloon Hong Kong

Abstract. This paper presents a novel explicit exploration information exchange mechanism for niche technique. In this framework, the whole population is divided into many sub-populations. The different sub-population communicates with each other. One sub-population exploration area does not be explored by others. Based on this framework, a multi-sub-swarm particle swarm optimization (MSSPSO) algorithm is implemented to test the thought. Five benchmark multimodal functions are used as test functions. The experimental results show that the proposed method has a stronger adaptive ability and a better performance for multimodal functions with respect to other niche techniques.

1 Introduction

The niche technique is a method that can make a population-based stochastic optimization algorithm to find multiple optimal solutions. According to its realization, the niche techniques can be classified sequential and parallel methods. In spite of Mahfound [1] had pointed out that the parallel niche technique is generally faster than the sequential one. Nevertheless, the sequential niche technique [2] does not allow one sub-population to explore a space searched by another sub-population, and this provides a good mechanism for information exchange. This exploration information exchange mechanism for the sequential niche technique still has its unique advantages. On the contrast, the parallel techniques [3]-[5] usually use some methods to avoid all individuals of the population converging to one optimum. In general, a complicated multimodal problem does not need to find all optima (including local and global optima). In this situation, those individuals converging in low peaks actually do not work for the optimization problem.

Recently, the niche particle swarm optimization attracts most researchers' attention. Niching particle swarm optimization proposed by R. Brits et al is the first

* This work was supported by the grants of the National Science Foundation of China, Nos. 60772130.

niche PSO technique [6]. It is the major drawback that the algorithm still need specify niche radius and other parameters in advance. In this paper, based on the proposed framework, we implement a multi-sub-swarm particle swarm optimization (MSSPSO) algorithm.

This paper is organized as follows. In section 2, we shall give some analysis to current niche techniques and propose a basic framework for explicit exploration information exchange. The related works will be presented in section 3. And in section 4, we will present the algorithm according to the proposed model. Section 5 will give the experimental results for a set of benchmark test functions. Section 6 draws some conclusions.

2 Basic Explicit Exploration Information Exchange Framework

In this paper, we propose a novel explicit exploration information exchange mechanism for niche technique. This mechanism can be applied into any population-based stochastic optimization algorithm. The basic explicit exploration information exchange framework can be described as follows: a population-based stochastic optimization algorithm is divided into many sub-populations, and each sub-population will explore one optimum for a multimodal problem. To accomplish this goal, firstly, the niche must be identified by sub-population. One sub-population must know which niche they exploring. Secondly, multi-sub-populations must exchange their exploration information explicitly, an effective information exchange will make one sub-population does not explore the niche belong to the others.

However, two problems will arise in this framework: how to exchange exploration information between different sub-population and how to identify an area (or niche)? We make some methods to implement this framework in this paper. In some sense, multimodal optimization can be viewed as a special constrained optimization problem. It is no necessary for a sub-population to explore a niche that had already been searched by another. Therefore, the penalty function [8] from the constrained optimization problem is introduced in multimodal optimization. This penalty will make the invader obliged to explore other niche. So, the penalty function used here can be viewed as an explicit exploration information exchange.

Then we must solve the next problem: How to identify a niche? In recent years, some researchers pay more attention to niche identification techniques (NIT) [9]. NIT can be used here to identify a niche. This paper uses a simple NIT: hill valley function. Besides, particle swarm optimization was used to implement the algorithm.

3 Hill Valley Function and Particle Swarm Optimization

The determination of the niche radius is generally a hard work existing in most niche methods. Ursem's hill valley function is the first method proposed in the literature [9] to identify the niche, which can be described as follow:

```

Hill_valley(ip,iq,samples)
minfit=min(fitness(ip),fitness(iq))
for j=1 to samples.length
Calculate point  $i_{interior}$  on the line between the points  $i_p$  and  $i_q$ 
    If (minfit> fitness( $i_{interior}$ ))
        return 1
    end if
end for
return 0
    
```

Fig. 1. Pseudo code of the hill valley function

where i_p and i_q are any two points in search space. Fig.2. just shows one dimensional function case. In fact, it can be easily extended to the case including arbitrary dimensions. With this function, the algorithm is able to determine whether i_p and i_q are to belong to one hill or not.

The samples array is generally used to calculate the interior points where the hill valley function computes the fitness of these samples. The points $i_{interior}$ can be calculated as:

$$i_{interior} = i_p + (i_q - i_p) \bullet samples [j] \tag{1}$$

where j is j th entry in the array. The upper boundary of j is the length of the samples, which is very important for the hill valley function. We refer to the length as sample rate (SR).

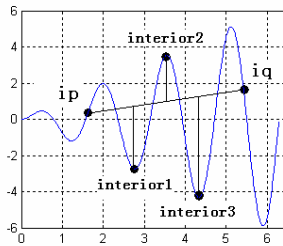


Fig. 2. The scheme for hill valley function

Particle swarm optimization (PSO) is a population-based stochastic search algorithm. The algorithm was firstly developed by Dr. Eberhart and Dr. Kennedy in 1995 [10], inspired by social behavior of bird flocking or fish schooling. The equations for the manipulation of the swarm can be written as:

$$V_{id} = W * V_{id} + C1 * rand() * (P_{id} - X_{id}) + C2 * rand() * (P_{gd} - X_{id}) \tag{2}$$

$$X_{id} = X_{id} + V_{id} \tag{3}$$

where $i = 1, 2, \dots, N$, W is called as inertia weight. $C1$ and $C2$ are positive constants, referred to as cognitive and social parameters, $\text{rand1} (*)$ and $\text{rand2} (*)$ are random numbers, respectively, uniformly distributed in $[0..1]$. The i th particle of the swarm can be represented by the D dimensional vector X_{id} , and the best particle in the swarm denoted by the index g , the best previous position of the i th particle is recorded and represented as P_{id} and the velocity of the i th particle is as V_{id} .

3 Multi-Sub-Swarm Particle Swarm Optimization Algorithm (MSSPSO)

In this paper, we use PSO to implement the framework proposed before. A multi-sub-swarm is employed to detect multi-solutions simultaneously, where every sub-swarm detects one solution. Considering that the most influential particle of a swarm is the globally best one, this work only uses the globally best particle of each sub-swarm to compete with each other. If two global best particles of different sub-swarm locate the same niche, we will compare their fitness, the larger is declared as the winner. The winner will own a marking niche, while the loser will be re-initialized in order to explore a new area.

In this work, we adopt the framework proposed before to solve the problem. The PSO algorithm actually has two influencing factors for a particle to move: the global best position of the swarm, and its private best position remembered from earlier. If we shift these two factors of a particle, then we can alter its tracking. In this paper, the algorithm does not directly change these two factors. On the contrary, the algorithm makes these particles lose their influence in their own sub-swarm. This is achieved by decreasing the fitness of a particle that invades another marking niche.

In this paper, we only use the simple penalty function. The modified fitness function of a particle that invades another niche must satisfy the following equation:

$$eval(x_n^i) = \begin{cases} f(x_n^i) & \text{if } \text{hill_valley}(x_n^i, x_k^{best}) = 1 \\ f(x_n^i) - p(x_n^i) & \text{otherwise} \end{cases} \tag{4}$$

In equation 4, x_n^i represent the i th particle of n th sub-swarm. x_k^{best} represent the best particle in k th sub-swarm. k is not equal n . $p(x_n^i)$ is penalty function, is constant in this paper.

The multi-sub-swarm niche PSO algorithm can be described as follows. The multi-sub-swarm was launched simultaneously. The niche in which the best particle of each sub-swarm is located is marked as that sub-swarm’s territory. The marking territory can be shifted, with the best particle moving to another niche. When two different sub-swarms occupy the same niche, the best particle of each sub-swarm competes with each other. The loser will be re-initialized. The basic algorithm can be described as follows:

Algorithm Multi-sub-swarm niche PSO algorithmCreate and initialize N sub-swarm of PSO algorithm

Repeat

For each sub-swarm,

If the best particle of different sub-swarms are located in the same niche

Compare their fitness: the smaller is marked as loser, the larger marked as

winner

Else

Mark the sub-swarm as winner

End if

Next

Reinitialize the loser sub-swarm; mark the winner's niche

For every particle and remembered particle position of each sub-swarm

If the particle invades another marked niche

Use equation 4 to decrease the fitness of the particle

End if

Next

Train each sub-swarm as original PSO algorithm

Until all sub-swarms converge or stopping condition is met

Fig. 3. Pseudo code of Multi-Sub-Swarm algorithm

The algorithm uses the hill valley function to determine whether two particles belong to one niche.

4 Experimental Results

In order to evaluate the effectiveness and the correctness of the method proposed in Section 3, a series of benchmark multimodal functions with different complexities are chosen. The optima of these functions are well understood.

The set of functions we have used in our experiment comprises seven functions, whose fitness landscapes exhibit an increasing complexity. Test functions F1-F5 were used by Beasley et al. [2]. These functions are defined as follows:

$$F 1(x) = \sin^6(5\pi x) \quad (5)$$

$$F 2(x) = (e^{-2 \log(2) \times (\frac{x-0.1}{0.8})^2}) \times \sin^6(5\pi x) \quad (6)$$

$$F 3(x) = \sin^6(5\pi(x^{3/4} - 0.05)) \quad (7)$$

$$F 4(x) = (e^{-2 \log(2) \times (\frac{x-0.1}{0.8})^2}) \times \sin^6(5\pi(x^{3/4} - 0.05)) \quad (8)$$

$$F 5(x, y) = 200 - (x^2 + y - 11)^2 - (x + y^2 - 7)^2 \quad (9)$$

Assume that Functions F1 to F4 are investigated in the range of [0, 1]. Function F5, the modified Himmelblau function, has 4 equal maxima with the value equal to 200. The range of x and y is in between [-6, +6].

The Performance Criteria for niching algorithm is listed below:

Maximum Peak Ratio: The maximum peak ratio is the sum of the fitness of the local optima identified by the niche technique divided by the sum of the fitness of the actual optima of a multimodal problem [4]. The maximum peak ratio can be defined as follows:

$$Peak \quad ratio = \frac{\sum_{i=1}^p f_i}{\sum_{i=1}^N f_i} \tag{10}$$

where p is the number of identified peaks, N is the total number of peaks, and f_i is the fitness of the i th peak.

Accuracy: The accuracy measures the closeness of the fittest solutions from all sub-swarms to all the known optima [9]. The definition is shown below:

$$accuracy = \frac{1}{|opt|} \left| \sum_{j=1}^{|opt|} |fit(opt_j) - fit(found_j)| \right| \tag{11}$$

where $|opt|$ gives the number of known global optima. $Fit(opt_j)$ is the fitness of j th optimum. $Fit(found_j)$ is the nearest particle.

Compared with the original PSO algorithm, only two extra parameters of our algorithm need to be tuned. The first is the sample array. The sample rate is set 2 and the sample array is defined as [0.01,0.09]. The other extra parameter is the number of sub-swarms. In general, we set this equal to the number of optima of the multimodal function. Other experimental parameters are similar to ordinary PSO algorithm. We assume that the inertia weight is from 0.9 to 0.4, $C1$ and $C2$ are set to 2.0, and V_{max} is set to the maximum range X_{max} . In addition, we let the maximum number of iterations be 100. In particular, for each functions used, 30 experiments are done with the same population size of each sub-swarm, and take the average result.

Five benchmark test multimodal functions were used to test the proposed algorithm and three other niche algorithms were used to compare their performance. These algorithms include restricted tournament selection (RTS)[5], Niching PSO algorithm [6] and adaptive niching PSO (ANPSO) [7]. In these experiments, the number of particles of each niche algorithm is set equal total particles used in MSSPSO for fair comparison. The population size of each other niche algorithm is set 30 for function F1-F5 respectively. The sub-swarms number of MSSPSO used in these experiments is set equal the number of optima and the population size of each sub-swarm is set 6, for function F1-F5.

Figure 5 (a)-(e) show the accuracy to corresponding function with the generation. Table 1 show maximum peak ratio and final accuracy of each algorithm. From Figure 5 and Table 1, we can see that the final accuracy of MSSPSO is much better than other niche algorithms. Besides, because of the explore information exchange, the population size of MSSPSO can be reduced hugely and the performance does not fall too many.

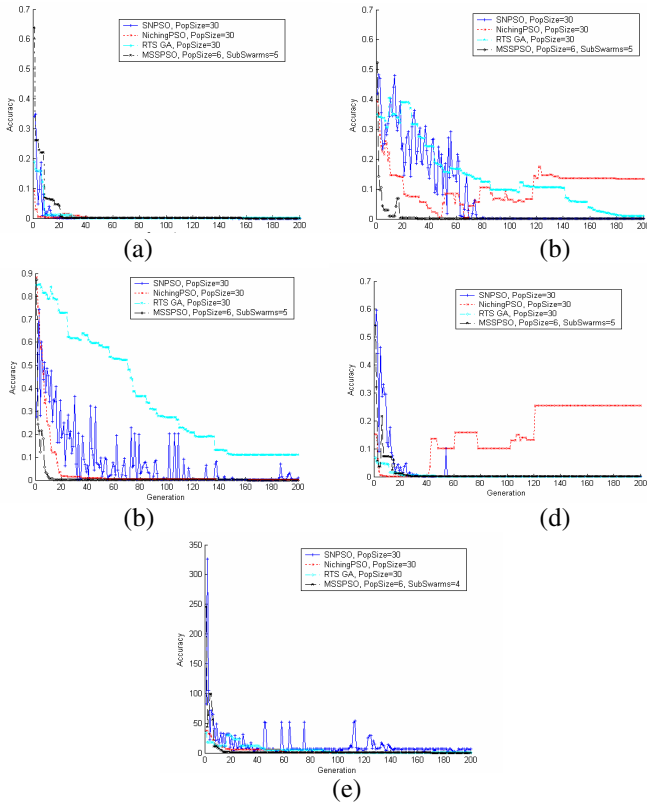


Fig. 4. a-e, The accuracy-generation for function F1-F5

Table 1. Maximum peak ratio and final accuracy

Test Function	Niche Algorithm	Maximum Peak Ratio	Final Accuracy
F1	MSSPSO	1	3.9968e-016
	Niching PSO	1	5.4194e-006
	SNPSO	1	1.3323e-016
	RTS	1	1.9027e-006
F2	MSSPSO	1	0.00060347
	Niching PSO	0.6231	0.13421
	SNPSO	1	0.00060347
F3	RTS	1	0.008623
	MSSPSO	1	3.9968e-016
	Niching PSO	1	8.3827e-005
	SNPSO	1	2.6645e-016
F4	RTS	1	0.022015
	MSSPSO	1	0.00052526
	Niching PSO	0.5863	0.25517
	SNPSO	1	0.00052526
F5	RTS	1	0.00037573
	MSSPSO	1	7.1054e-015
	Niching PSO	1	0.10893
	SNPSO	0.9632	5.2808
	RTS	0.9991	0.34258

5 Conclusions and Future Works

This paper has presented an explicit exploration information exchange framework for niche technique. Based on this framework, we implemented a novel multi-sub-swarm PSO (MSSPSO) algorithm. Five benchmark multimodal functions were used as test functions. The experimental results show that the proposed method has a good performance and strong adaptive ability.

Of course, this is only a preliminary work. The proposed framework still needs further expansion. The hill valley function used in this paper is very simple, how to analyze the topology of an optimization problem is the next research focus. Moreover, how to apply the constrained optimization technique to multimodal optimization is need further research. Especially, how to choose an appropriate penalty function for proposed framework.

References

1. Mahfoud, S.W.: A Comparison of Parallel and Sequential Niching Methods. In: Proceedings of the Sixth International Conference on Genetic Algorithms, pp. 136–143 (1995)
2. Beasley, D., Bull, D.R., Martin, R.R.: A Sequential Niching Technique for Multimodal Function Optimization. *Evolutionary Computation* 1(2), 101–125 (1993)
3. Goldberg, D.E., Richardson, J.: Genetic Algorithms with Sharing for Multimodal Function Optimization. In: Grefenstette, J.J. (ed.) Proc. 2nd Int. Conf. Genetic Algorithms, pp. 41–49. Lawrence Erlbaum, Hillsdale (1987)
4. Mahfoud, S.W.: Crowding Preselection Revisited. *Parallel Problem Solving from Nature*, vol. 2, pp. 27–36. Elsevier, Amsterdam (1992)
5. Harik, G.R.: Finding Multimodal Solutions Using Restricted Tournament Selection. In: Proceedings of the Sixth International Conference on Genetic Algorithms (1995)
6. Brits, R., Engelbrecht, A.P., van den Bergh, F.: A Niching Particle Swarm Optimizer. In: Conference on Simulated Evolution and Learning, Singapore (2002)
7. Bird, S., Li, X.D.: Adaptively Choosing Niching Parameters in a PSO. In: Proceedings of the 8th annual conference on Genetic and evolutionary computation, Washington, USA, vol. 1, pp. 3–9 (2006)
8. Yeniy, Ö.: Penalty Function Methods for Constrained Optimization with Genetic Algorithms. *Mathematical and Computational Applications* 10(1), 45–56 (2005)
9. Ursem, R.K.: Multinational Evolutionary Algorithms. In: Proceedings of Congress of Evolutionary Computation, vol. 3, pp. 1633–1640 (1999)
10. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: Proc. Of IEEE International Conference on Neural Networks (ICNN), Perth, Australia, vol. 4, pp. 1942–1948 (1995)

The Research of Using Ant Colony Algorithm in Solving Sequencing Problem of Mixed Model Assembly Lines with Multi-objectives*

Qiong Zhu, Lihui Wu, and Jie Zhang

Computer Integrated Manufacturing Research Institute, School of Mechanical Engineering,
Shanghai Jiao Tong University, Shanghai, 200240, China
zhangjie@sjtu.edu.cn

Abstract. Mixed model assembly lines (MMAL) sequencing problem is a typical NP-hard problem. It's important to search for an optimal sequence to maximize the efficiency of production lines. Therefore this work proposes a new mathematic programming model for measuring the efficiency of MMAL. A modified ant colony algorithm (MACA) is developed to determine a sequence which optimizes the objective function with elitist strategy. The experimental results indicate the applicability of the proposed objective function and availability of the algorithm in solving this problem.

Keywords: Mixed Model Assembly Line (MMAL), sequencing; ant colony algorithm, elitist strategy.

1 Introduction

MMAL play a key role in Just in Time (JIT) based production environment. MMAL sequencing is recognized as an important work to improve the utilizing efficiency of production line. During the last four decades, MMAL optimal objectives have been translated into different goals, two goals of which are the original ones. Dar-El and Cucuy proposed GOAL 1 to minimize idle time of conveyers [1]. Xiaobo ,Zhou et al. delivered GOAL 2 to keep the constant usage of every part [2]. GOAL 2 has been applied to the real production environment of Toyota automobile assembly line and approved to support the Just in Time concept in Toyota production system [3]. Other goals are extension of the former two goals. For example, Dar-El advanced the goal of minimizing throughput time by reducing the time wasted on waiting for the jobs [4], which is obviously the extension of GOAL 1. Kubiak presented the objective to keep the constant rate of every model fed into the assembly line [5]. In fact, models mentioned in this example are parts of the finished product, so this goal is just a branch of GOAL 2.

The rest of this work will focus on the objective of maximizing the efficiency of assembly lines, which is obtained by integrating GOAL1 and GOAL2 together via

* This paper is supported by the National Natural Science Foundation of China, No. 50575137 and the National High-Tech Research and Development Plan of China(863 Program), No.2007AA04Z109.

adding time element, and for which modified an ant colony algorithm has been proposed to obtain improved solutions.

The organization of this paper is as follows. Section 1 is a brief review on the mixed model sequencing problem and an introduction of the related research background. In section 2, a mathematical programming model for the MMAL sequencing problem with new objective is presented in details. In section 3, the ant colony algorithm with elitist strategy is proposed to solve the sequencing problem. Section 4 discusses the experimental result of the proposed approaches. Finally, the conclusions are made.

2 MMAL Sequencing Problem

Mixed model assembly line sequencing problem is a complex combinatorial optimize problem. One theme that addressed in some of the former approaches is that the mixed model sequencing problem is to feed a group of models repetitively. This paper defines a group of models as the collection of different models which are to be assembled concurrently and use the parts interchangeably [6].

Here gives the definition of notations:

- $M = \{1, \dots, M\}$ set of models;
- $K = \{1, \dots, K\}$ set of workstations on the assembly line;
- $t_{p_{mk}}$ total processing time of model $m(\in M)$ on station $k(\in K)$;
- $P = \{1, \dots, P\}$ set of parts on the assembly line;
- $N = \{1, \dots, N\}$ set of the units;
- $D = \{D_1, \dots, D_M\}$ the demand amount of different models;
- $\pi = \{\pi(1), \dots, \pi(N)\}$ $\pi(n)(n \in N)$ is the n th unit in the sequence π .
- b_{mpk} is the requirement for part $p(\in P)$ for assembling model m one unit at workstation $k(\in K)$;

In former literatures the ideal usage rate of every part is defined as the requirement for part $p(\in P)$ per unit feeding into the assembly line in formulation (1).

$$\alpha_p = \frac{\sum_N b_{mpk}}{N} \tag{1}$$

This paper considers an assembly line with multiple workstations and certain process time, the ideal consuming rate is defined as follow,

$$\alpha_{pk} = \frac{\sum_N b_{mpk}}{NT_N} \tag{2}$$

Where: T_N is the time span for the process of the entire N unit.

The objective of this paper is to find a sequence which minimizes

$$G(t) = \min \sum_{n=1}^N \sum_{m=1}^M \sum_{k=1}^K \sum_{p=1}^P (nt\alpha_{pk} - b_{mpk} - \beta_{n-1,p,k})^2 x_{nm} \tag{3}$$

Where:

$\beta_{n-1,p,k}$ is the amount of part $p(\in P)$ used in workstation $k(\in K)$ for the former $n - 1$ units;

$x_{nm} \in \{0,1\}$, 1, if the model $m(\in M)$ is at the position $n(\in N)$, 0 otherwise.

The objective function Eq. (3) is differ from the traditional GOAL2 with characteristic of adding time factor to the ideal consuming rate, this paper is aimed to minimize the objective function so as to obtain a sequence which maximizes the efficiency.

3 Modified ant Colony Algorithm

Ant Colony Optimization (ACO) algorithm is an artificial intelligence based meta-heuristic algorithms with splendid performance in solving combinatorial optimize problems, including Traveling Salesman Problem (TSP) in [7], Quadratic Assignment Problem in [8], and Vehicle Routing Problem in [9], et al. MMAL sequencing is one of the typical complex combinatorial optimization problems. Therefore the ACO algorithm is introduced to the core problem of MMAL sequencing. In the following, a special search model is illustrated and the MACA is presented at the same time.

3.1 A Graph Model for MMAL Sequencing

The MMAL sequencing problem can be described as an undirected graph $G = (V, E)$, where $V = \{start, Unit1, Unit2, \dots, end\}$ is the nodes set in the graph G (Fig. 1), and $E = \{e_{ij} | i, j \in V\}$ is the arc set connect different nodes in V . $Unit i$ stands for

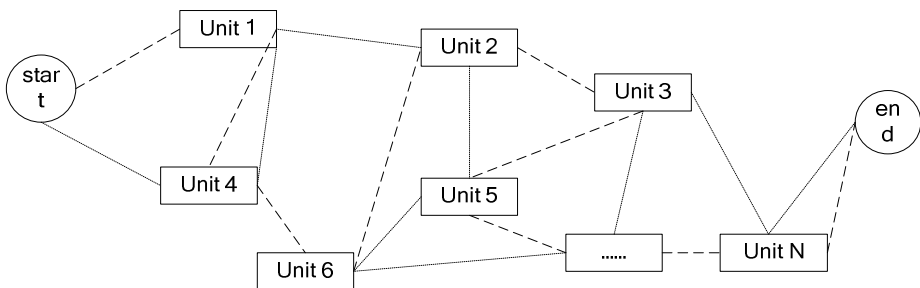


Fig. 1. The undirected graph G gives two different paths from the start point to the end point with dashed lines and dotted dash lines

different product models. The two nodes named *start* and *end* are the virtual nodes of beginning and end-point in MMAL sequencing. Sequencing different product models is seemed as the ants find the best tour route to cover all the nodes (units) which minimize the proposed objective function of (1). Fig. 1 gives two different routes from the start point to the end point with dashed lines and dotted dash lines.

3.2 Ant Colony Algorithm with Elitist Strategy

In the following, a special MACA algorithm with elitist strategy is designed to exploit optimization sequence for MMAL is introduced.

1) Create ants: the amount of ants is M , each ant searches for a feasible path pass through all the nodes in the graph G .

- a) Starting point: *start* , end point: *end*
- b) Initial pheromone > 0 , but very small

2) Find solutions. The ants choose the next point of unit $n \in N$ according to the state transfer possibility formula (4):

$$P_{ij}(t) = \frac{\tau_{ij}(t)^a \eta_{ij}(t)^b}{\sum_{j \in \text{allowed nodes}} \tau_{ij}(t)^a \eta_{ij}(t)^b} \tag{4}$$

Where i, j are the notation of different units, $\tau_{ij}(t)$ is the quality of pheromone at time t , $\eta_{ij}(t)$ is the heuristic value at time t , here $\eta_{ij}(t) = 1/t_{ij}$, a, b are constant representing the important degree of pheromone quantity and heuristic value.

3) Check if all the points are covered or not, if yes evaluate the sequences (paths) with the objective function (3), otherwise back to 2);

4) Update pheromone trail using elitist strategy. According to the evaluation result the special ant which finds the shortest path (i.e. best evaluation result) in its iteration is defined as elitist ant. To make sure that the shortest path obtains more attractive than the other paths, the elitist ant should place more pheromone on the way back then others. The pheromone update formula is defined in below:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij} + \Delta\tau_{ij}^* \tag{5}$$

Where: $\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k$.

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & \text{if the ant } k \text{ use the path } (i, j) \\ 0 & \text{otherwise} \end{cases}$$

$$\Delta\tau_{ij}^* = \begin{cases} \sigma \cdot \frac{Q}{L^*} & \text{if edge}(i, j) \text{ is part of the shortest path} \\ 0 & \text{otherwise} \end{cases}$$

Where $\Delta\tau_{ij}^*$ is the extra pheromone added to edge (i, j) , σ is the amount of the elitist ant, L^* denotes the evaluate result of optimal solution.

6) Until the terminal station satisfied, back to 2)

4 Experimental Results

Suppose there are 3 workstations $K = \{1, 2, 3\}$ within the assembly line producing three product models $\{A, B, C\}$. The detailed processing information for each operation is listed in Table 1.1 in the format: *station number(processing time, the amount of needed part)*. In the following, a set of MMAL sequencing instances have been employed to do the test in table 2. The number of feasible solutions for each problem is given in the forth column synchronously. Goal chasing algorithm and the basic ACO algorithm are applied to compare with MACA approach.

Table 1. The detailed processing information for product models

model	1	2	3	4	5
A	1(1,2)	0	2(4,1)	0	3(2,3)
B	1(2,2)	1(2,4)	2(3,1)	0	3(2,4)
C	0	1(3,7)	2(1,8)	2(3,3)	3(1,1)

Table 2. Problem sets

problem	n	D	No. of feasible solutions
1	9	(4, 3, 2)	1260
2	10	(5, 3, 2)	2520
3	10	(4, 4, 2)	3150
4	11	(6, 3, 2)	4620
5	12	(6, 4, 2)	13860

In MACA the parameters have been set to: $a = 2, b = 3, \rho = 0.2, Q = 100$.

Table 3 shows the performance of three algorithms mentioned above. Comparing with the other two algorithms, the numbers of better solution obtained by MACA are 3, 3 and 2, the proportions are 60%, and 60% respectively.

Table 3. The optimal solution of different algorithms

No.	Goal Chasing		ACO		MACA	
	Min	Sequence	Min	Sequence	Min	Sequence
1	25.2	BACABACBA	25.2	ABCABCABA	20.5	ABCABACBA
2	24.1	ABCAABACBA	24.1	ABCAABACBA	24.1	ABCAABACBA
3	29.7	BACABABCAB	29.3	BACABABCAB	27.3	ABCABABCAB
4	33.1	ABCAABACAAB	30.9	CABAABACABA	24.3	ABACABACABA
5	29.6	ABACBAABACBA	29.6	ABACBAABACBA	29.6	ABACBAABACBA

As expected MACA algorithm is able to search for solutions much better than the other algorithms. This is confirmed the availability of improvements in MACA by integrating elitist strategy to the pheromone refreshing process. This strategy strengthened the attractive of the best route by awarding elitist route extra pheromone, therefore to MACA can avoid stagnation caused by equal distribution of the pheromone. By means of building a MACA algorithm particularly for MMAL sequencing, the results verify that the performance of MACA is significantly better than the other two algorithms.

5 Conclusions

In this paper, a new objective function for MMAL is formulated to measure the efficiency with consideration of both total circle time and consuming rate of every part. And a modified ant colony algorithm is proposed as a perfect way to solve this kind of combinatorial optimal problem. The test results indicate the applicability of the proposed objective function and availability of the algorithm in solving those problems. Directions for further research will concentrate on the proof-test of the proposed approaches in more practical cases, i.e. considering about more complicated situations, for example, each operation may fix on amount of different parts.

References

1. Dar-El, E.M., Cucuy, S.: Optimal Mixed-model Sequencing for Balanced Assembly Lines. *Omega* 5, 333–342 (1977)
2. Xiaobo, Z., Zhou, Z., Asres, A.: A Note on Toyota's Goal of Sequencing Mixed Models on an Assembly Line. *Computers & Industrial Engineering* 36, 57–65 (1999)
3. Becker, C., Scholl, A.: A Survey on Problems and Methods in Generalized Assembly Line Balancing. *European Journal of Operational Research* 168, 694–715 (2006)
4. Dar-El, E.M.: Mixed-model Assembly Line Sequencing Problems. *Omega* 6, 313–323 (1978)
5. Kubiak, W.: Minimizing Variation of Production Rates in Just-in-time Systems: A Survey. *European Journal of Operational Research* 66, 259–271 (1993)
6. Karabati, S., Sayin, S.: Assembly Line Balancing in a Mixed-model Sequencing Environment with Synchronous transfers. *European Journal of OR* 149, 417–429 (2003)
7. Dorigo, M., Gambardella, L.M.: Ant Colonies for the Travelling Salesman Problem. *Bio-systems* 43, 73–81 (1997)
8. Shyu, S.J., Lin, B.M.T., Hsiao, T.S.: Ant Colony Optimization for the Cell Assignment Problem in PCS Networks. *Computers & Operations Research* 33, 1713–1740 (2006)
9. Gagne, C., Gravel, M., Price, W.L.: Solving Real Car Sequencing Problems with Ant Colony Optimization. *European Journal of Operational Research* 174, 1427–1448 (2006)

A New Weighted Support Vector Machine for Regression and Its Parameters Optimization*

Liquan Mei and Shujuan Zhang

School of Science, Xi'an Jiaotong University
Xi'an, 710049, P.R. China
lqmei@mail.xjtu.edu.cn,
yuliting@stu.xjtu.edu.cn

Abstract. In this paper what we study is twofold. Firstly a new weighted support vector machine(WSVM) is introduced, in which different roles of samples are considered. It better processes the singularity in the sample than SVM. Secondly based on grid search and the solution path algorithm, a new algorithm is given to quickly find the optimum parameters. Numerical results show the effectiveness and the stability of the algorithm.

Keywords: WSVM, singularity, regression, parameters optimization.

1 Introduction

SVM is based on SRM (Structural Risk Minimization). Through the kernel technology, SVM can easily transform nonlinear problem in sample-space to linear problem in high-space. It uses the similarity among the samples in high-space to find a hyperplane with largest margin that can correctly separate the sample. Regression problems can be processed like classification problems because the regression problem solved by SVM is a special classification problem^[1].

Following formula is very important for SRM^[2]

$$R(f) \leq R_{\text{emf}}(f) + \sqrt{\frac{h(\ln(2l/h) + l) - \ln \frac{7}{4}}{l}}. \quad (1)$$

In this formula, $R(f)$ is the expected risk, $R_{\text{emf}}(f)$ is the empirical risk, the second term on the right of (1) is the confidence interval, h is the VC dimension of the fitted functional space, and l is the number of training samples. SRM is different from ERM (Empirical Risk Minimization) which only minimizes $R_{\text{emf}}(f)$. Minimizing $R_{\text{emf}}(f)$ can fit the sample well while easily lead to over-fitting and minimizing the confidence interval can decrease the test risk. Over-fitting problem can be avoided when the two targets are minimized at the same time. When h increases, the confidence interval increases but the $R_{\text{emf}}(f)$ is decreases, so we can get a good model by choosing a reasonable h . There have been many results in theory and application about SVM since it was proposed by

* The project is supported by NSF of China(10471109) .

N.Vapnik in 1990s, such as in the pattern recognition, signal processing(Vapnik, Golowich,& Smola,1996), time series prediction(Muller et al.,1997), and neural decoding(Shpigelman, Crammer, Paz, Vaadia,& Singer,2004)and so on. It has been a popular topic in recent years.

WSVM performs better than ε -SVM because the different roles of the samples are considered. But generally in the sample-space the Euclidean distance between training samples and test samples is chosen as the weight of WSVM. It takes very long CPU time because it has to train again in order to forecast a new test sample.

According to the above discussion, a new weight is introduced in this article. It inherits the advantage of general WSVM and only needs short CPU time. As is well known, the performance of SVM depends on the parameters, but it is difficult to choose suitable parameters for different problems. Much work has been done for this, such as [1], [4]. In order to finish this work we will introduce an algorithm which is proved fast and effective through numerical tests.

2 Weighted Support Vectors Machine

Consider

$$\begin{aligned} \min_{w, \xi, \xi^*} & \frac{1}{2} w^T \cdot w + C \sum_{i=1}^l w_i (\xi_i + \xi_i^*), & (2) \\ \text{s.t} & f(x_i) - y_i \leq \xi_i + \varepsilon, \quad \xi_i > 0, \quad i = 1 \cdots l, \\ & y_i - f(x_i) \leq \xi_i^* + \varepsilon, \quad \xi_i^* > 0, \quad i = 1 \cdots l, \end{aligned}$$

where $f(x) = \langle w, \varphi(x) \rangle + b$. We solve the following dual problem of (2) instead of directly solving (2)

$$\begin{aligned} \min_{a, a^*} w(a, a^*) &= \frac{1}{2} \sum_{i,j=1}^l (a_i - a_i^*)(a_j - a_j^*)K(x_i, x_j) \\ &+ \varepsilon \sum_{i=1}^l (a_i + a_i^*) - \sum_{i=1}^l (a_i - a_i^*)y_i & (3) \\ \text{s.t} & \sum_{i=1}^l (a_i - a_i^*) = 0, \quad 0 \leq a_i, a_i^* \leq C * w_i, \quad i = 1 \cdots l, \end{aligned}$$

where w_i is the weight for the i th sample, $0 \leq w_i \leq 1$. Good fitting function can be got by choosing appropriate functional space. But when there are some singularities in the training sample or when there is noise with heteroscedasticity, for example, the noise adding to the first section of samples is $N(0, 0.8)$, the noise for the remaining samples is $N(0, 0.1)$, the standard ε -SVM can't get reasonable fitted function, but WSVM can. The numerical test shows that WSVM can powerfully decrease the effect of the singularity, see examples in section 4 .

Here the weight of WSVM is taken as follows^[3]:

$$e_i = |y_i - f(x_i)|, \quad y_i \text{ is the true value, } f(x_i) \text{ is the fitting value,}$$

$$e1_i = \begin{cases} 0 & \text{if } e_i \leq \eta \\ \frac{e_i}{\eta} & \text{if } e_i > \eta \end{cases}, \quad w_i = \begin{cases} 1 & \text{if } e1_i = 0 \\ \frac{1}{1+(e1_i)^{\frac{1}{p-1}}} & \text{if } e1_i \neq 0 \end{cases}, \quad (4)$$

where η, p are prefixed positive constants. The new weight doesn't relate to the test samples, so it consumes shorter CPU time than general WSVM does. Table 1 in section 4 shows this WSVM is superior to ε -SVM when there are some singularities in the training samples. But there should be a premise if this WSVM performs well. The premise is that the number of the singularities of the two sides of function curve should be similar, and the distance of singularities of the two sides to the function curve should be similar, or the fitted curve will direct towards one side with more and larger singularities. This makes the algorithm short of bad generalization. Table 2 in section 4 shows this fact.

Based on the above discussion, this article propose a new type weight to overcome the above shortcoming but inherite the advantage of above WSVM. The new weight is as follows:

(1) Set

$$E^+ = \{x_i | y_i - f(x_i) > \eta\},$$

$$E^- = \{x_i | y_i - f(x_i) < -\eta\},$$

$$E^c = \{x_i | |y_i - f(x_i)| \leq \eta\}.$$

(2)

$$sum^+ = \sum_{i=1}^{l^+} (y_i - f(x_i))^2 + \alpha, \quad x_i \in E^+,$$

$$sum^- = \sum_{i=1}^{l^-} (y_i - f(x_i))^2 + \alpha, \quad x_i \in E^-.$$

(3)

$$w_i = \begin{cases} \left(\frac{|E^-|+1}{|E^-|+|E^+|+2}\right)^{p1} * \left(\frac{1}{sum^+}\right)^{p2} * \frac{1}{1+\left(\frac{|y_i-f(x_i)|}{\eta}\right)^{p3}} & x_i \in E^+ \\ 1 & x_i \in E^c \\ \left(\frac{|E^+|+1}{|E^-|+|E^+|+2}\right)^{p1} * \left(\frac{1}{sum^-}\right)^{p2} * \frac{1}{1+\left(\frac{|y_i-f(x_i)|}{\eta}\right)^{p3}} & x_i \in E^- \end{cases} \quad (5)$$

where $\alpha, \eta, p1, p2, p3$ are prefixed positive constants, and $|E^+|, |E^-|$ means the number of elements in the corresponding space. The deviation of singularities of two sides of the functional curve is considered in this new weight. Table 1 and 2 displays that WSVM using weight in (5) has advantage over WSVM using weight in (4). Numerical result shows that WSVM introduced in (5) is less sensitive to C, that is of the better stability. Following WSVM using weight in (4) is called WSVM and WSVM using weight in (5) our WSVM for short.

3 Parameter Optimization

As in any other smoothing problem, it is critical to choose the regularization parameter C in order to get a good fitted model that performs well on future data(i.e.,small generalization error). Numerical test shows RBF is good kernel when solving nonlinear regression problems by WSVM. RBF is as follows:

$$K(x, x_i) = \exp\left(\frac{|x - x_i|^2}{2\sigma^2}\right)$$

where σ^2 is a parameter that plays an important role in the generalization ability of the algorithm.

Now parameters C, σ^2, ε need to be optimized in WSVM regression. Three ways: cross-validate, grid search and solution path, were used for parameter optimization. The defect of cross-validate is time-consuming, that of grid search artificially choosing the interval which the grid is in and the search step. In order to overcome the above defects a new algorithm using GCV standard is improved in this article. GCV is good at avoiding over-fitting. We compare GCV with other standard here, and the results in Fig.1-2 show that GCV is a better standard than the others.

In WSVM, ε can be easily chosen, so there is no discussion for it in this article. We only focus on the selection of C and σ^2 . The algorithm for parameter optimization is the integration of fast solution path and grid search.

3.1 Use Fast Solution Path to Choose Interval

Gunter and Zhu^[4] deeply discussed SVM and proposed a new algorithm, which is called solution path algorithm to find satisfying parameters of SVM. It computed the entire solution path of the SVR, then chose the optimal parameter that minimized GCV, see [4] for details. We did many numerical tests using this algorithm, and found some shortcomings. Following is our improvement of solution path:

(1) Initialization. It does not need to start from $\lambda = +\infty$ because the algorithm bad fits training sample in this case. Choosing a small λ such as $\lambda = e^4$ is reasonable, then decrease λ until df is more than 4. The solution path algorithm proceeds from here.

(2) At every updating step the algorithm in [4] chose the largest λ that caused an event. Numerical tests show the algorithm takes too long CPU time for too small change of λ . In this article, we choose the middle value of λ between the largest λ and the smallest λ causing an event. Numerical test displays this improvement is effective and the final λ is reasonable.

(3) Set

$$GCV = \begin{cases} \frac{\sum_{i=1}^l (y_i - f(x_i))^2}{1 - df/l} & \text{if } df \neq l \\ l * \sum_{i=1}^l (y_i - f(x_i))^2 & \text{if } df = l \end{cases} \tag{6}$$

where df is the number of the samples that happen to be on the border of the ε pipeline^[1]. After running the above improved solution path we can get an initial parameter, then we use following grid search to further optimize it.

3.2 Use Grid Search to Choose Optimal Parameter

Through observing many numerical tests, we find that σ^2 doesn't relate to C . The fitting trend is similar when C is fixed while σ^2 changes, so this article uses an algorithm as follows:

Algorithm 1

Initialize GCV, C

σ^2 loop

Standard ε -SVM;(or WSVM)

Calculate the standard(for example GCV);

σ^2 loop end

Choose σ^2 that minimizes the standard. The chosen σ^2 is used in the following steps.

C loop

Standard ε -SVM;(or WSVM)

Calculate the standard(for example GCV);;

C loop end

Choose C that minimizes standard.

The usual standards are as follows:

(1) ERM(Empirical Risk Minimization). It means $\frac{1}{l} \sum_{i=1}^l (y_i - f(x_i))$, x_i is the training sample. Using ERM will lead to over-fitting easily.

(2)Cross-validate. This algorithm divides the sample into several parts. Every time it chooses one part as the test samples and the remaining as the training samples, then it trains and computes the test error using the training fitted function. The sum of the test error is got after some loop steps. It is a time-consuming algorithm.

(3) GCV(General Cross Validate). GCV standard shows the sparsity of SVM again. When it has reached required fitting accuracy to some extent, the less support vector the result has, the stronger the generalization ability the fitting function is.

Algorithm 1 take shorter CPU time than double cycle^[5]. And more reasonable parameters can be found using Algorithm 1 when we start algorithm with small C.

4 Numerical Results

First, the samples generated from simulating are discussed to test our algorithms. Then trial samples are considered to validate the robustness of the algorithms. We runs all algorithms on plat of matlab 6.5.

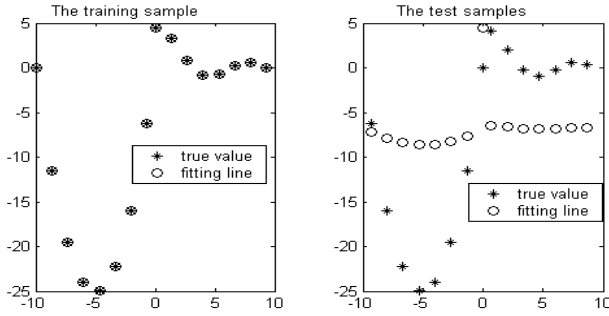


Fig. 1. Over-fitting happens using ERM standard

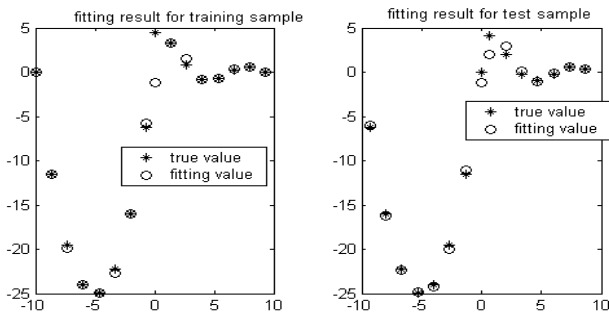


Fig. 2. Using GCV standard can avoids over-fitting

Example 1. Consider piecewise function

$$f(x) = \begin{cases} 10 * x + x^2 & x \leq 0 \\ 4.5 * \sin(x) & x > 0, \end{cases}$$

We take RBF as the kernel of SVM. Parameters σ^2 and C are chosen using algorithm in section 3. Numerical results are summarized in Fig 1, and Fig 2.

Example 2. The function is $y = \text{sinc}(x)$, kernel is RBF, $\sigma^2 = 0.7$. First we choose 27 equally spaced points in $(-\pi, \pi)$, then we add 10 to $y(3)$ and $y(27)$ respectively and -10 to $y(10)$ and $y(17)$ respectively. Above points are training samples. we choose 27 other equally spaced points as test samples. Part of the results is displayed in Table 1 . All values in Table 1 are the test error $\sum_{i=1}^t (y_i - f(x_i))^2$, x_i is the test sample. From the numerical results, we know that our WSVM is obviously better than WSVM. Our WSVM is more insensitive to C than WSVM, namely it is of better robustness.

Example 3. Function and test samples are the same as Example 2. Its training samples are as follows: First 27 equally spaced points in $(-\pi, \pi)$ are created; secondly a variable is added to $y(3)$; thirdly other two variables are subtracted from

Table 1. Test error of example 2 using different algorithm

Order	C	Error of SVM	Error of WSVM	Error of our WSVM
1	1	0.00097582	0.00026476	0.00026537
2	5	0.024066	0.0050019	0.00050373
3	9	0.078103	0.00064415	0.00065245
4	13	0.14867	0.00082422	0.00084527
6	17	0.2012	0.00092923	0.00016678
8	25	0.5825	0.00086831	0.00015604
9	67	4.3413	0.012685	0.00011882
10	69	4.6071	0.032469	0.00011756
11	71	4.8809	0.064327	0.00011645
12	73	5.1635	0.19959	0.00011549
mean		1.7525	0.0091	$2.2085e^{-4}$

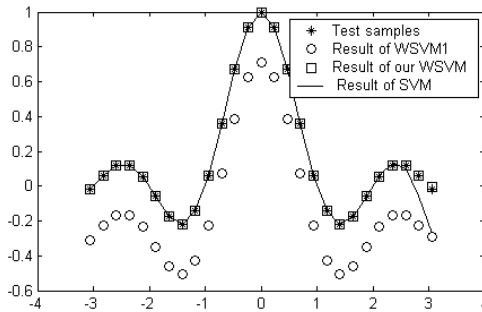


Fig. 3. Compare the result got by different methods for example 3

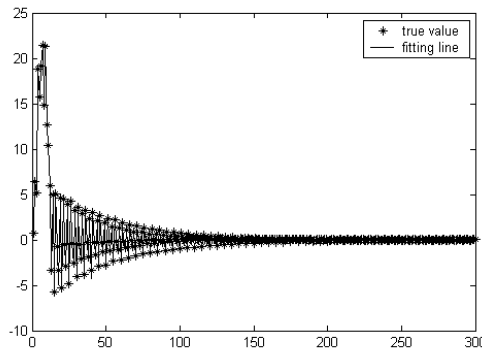


Fig. 4. Numerical results for structural vibration of cylinder shell

$y(10)$ and $y(27)$ respectively. The three variables are random variables larger than 10. There is deviation of singularities of two side of the function curve in these samples. Numerical results show that WSVM performs poorly. Its result

Table 2. Test error of example 3 using different algorithm

Order	C	Error of SVM	Error of WSVM	Error of our WSVM
1	1	5.3848e-005	0.07364	2.805e-005
2	1.5	0.00051058	0.07363	1.9763e-005
3	2	0.001446	0.082065	1.7623e-005
4	2.5	0.0028877	0.082063	1.3348e-005
5	3	0.0047962	0.082061	9.7466e-006
6	3.5	0.0071848	0.082061	8.5384e-006
mean		0.0028	0.0793	$1.6178e^{-5}$

directs towards the one side with more and larger singularities. In this case our WSVM is obviously advantaged, see Fig.3 and Table 2.

Example 4. Considering practical trial data which generated from structural vibration of cylinder shell, we implement regression analysis using our WSVM algorithm. The numerical results are shown in Fig.4.

Numerical tests show that our WSVM is robust, effective and can be widely applied to engineering problems.

References

1. Deng, N., Tian, Y.: A New Data Mining Method: Support Vector Machine. Science Press, Beijing
2. Vapnik, V.N.: The Nature of Statistical Learning Theory. Springer, New York (1995)
3. Guo, G.: Support Vector Regression: Algorithm and Improving. Thesis of Xi'an Jiaotong university (2004)
4. Gunter, L., Zhu: Efficient Computation and Model Selection for the Support Vector Regression. *Neural Computation* 19(6), 1633–1655 (2007)
5. Zhao, Y., Liu, H.: A New Method for SVM Hyper-parameters Optimization. *Journal of Chinese Computer System* 29(1), 102–105 (2008)

A Novel Spike Sorting Method Based on Semi-supervised Learning*

Guo-Zhu Wen^{1,2} and De-Shuang Huang¹

¹ Intelligent Computing Lab, Institute of Intelligent Machines,
Chinese Academy of Sciences, China

² Department of Automation, University of Science and Technology of China, China
dshuang@iim.ac.cn

Abstract. Spike sorting is a prerequisite technique for neural coding and decoding research. So far many manual and automatic spike sorting approaches have been proposed, however this issue is still a challenge due to high time consumption of human or unreliability of machine. In this study, a semi-supervised spike sorting framework was proposed, in which three clustering or learning components, i.e. self-organizing map (SOM), manifold learning and affinity propagation clustering (APC) work cooperatively. The SOM serves as the technique for data reduction, as well as the visualization of cluster structure, which enabled the operator's "supervised" intervention. The manifold learning technique describes the intrinsic structure underlain by SOM grid, and yields the similarity matrix as output. The affinity propagation clustering takes as input the yielded similarity matrix, followed by the human's guidance, and outputs the reasonable clustering result. The advantage of our framework is the efficient combination of traditional and novel approaches, wherein the semi-supervised intervention is embedded appropriately to guide the clustering process. The practical experiments' results show that the proposed approach performed very well.

Keywords: Semi-supervised Clustering, Spike Sorting, Self-organizing Map, Manifold Learning; Affinity Propagation.

1 Introduction

In the last few decades, researchers have been trying to reveal the correlation between human intelligence and the underlying neural mechanism within our brain. A feasible way is to record the event potential of the large-scale neuronal ensembles followed by analyzing the pattern within the potential serials. Now the technique of multi-channel electrodes allows for the simultaneous recording of many neurons, and makes it possible to analyze the spike pattern of each recorded neuron [1]. Generally, the widely used extracellular recording techniques record many neurons around the microelectrode simultaneously, thus the action potential recorded by a single microelectrode may contain several neurons' activities. Obviously the isolation and identification of a single

* This work was supported by the grant of the Guide Project of Innovative Base of Chinese Academy of Sciences (CAS), No.KSCX1-YW-R-30, and the grant of Oversea Outstanding Scholars Fund of CAS, No.2005-1-18.

neuron's signal from the mixed ones, the process that is referred to as spike sorting (or spike classification), is the prerequisite of investigating information processing mechanisms within the nervous system [2].

Generally the procedure of spike sorting is divided into three stages [3]: (1) Spike detection. (2) Feature extraction. (3) Clustering of the spikes in a specific feature space. In this stage, the spikes having the similar features were partitioned as a group representing a putative neuron.

Due to the massive data and high dimension feature space to be processed, the manual spike sorting is time-consuming. Thus, many literatures proposed using the automatic spike sorting algorithm so as to make the process faster and more objective [2][3][4]. However, most of the automatic methods are based on the priori assumption about the distribution of data, such as Gaussian distribution, or t-distribution. Unfortunately, this priori assumption cannot accurately capture the data's statistical characteristics, especially when the Signal-Noise-Ratio (SNR) is low. The violation of supposed distribution will bias most automatic clustering algorithms, and lead to unreliable results. Therefore many laboratories prefer to cluster this kind of data manually. Nevertheless, besides the drawback of time-consuming, the manual operation is subjective and information-limited. First, different operators may give different results, which cannot be evaluated by quantification. Some experts may give a perfect sorting result, but most operators can't. Second, people can only view the samples from at most three-dimension feature space although many features are available. Therefore the information loss is inevitable. Sometimes, the clusters are distinct in high dimension space, but are overlapping in the low dimension space. Instead, the computer algorithm can work in the high dimension feature space so that more information is considered. In order to integrate the advantages of automatic and manual method, some researchers proposed using semi-automatic algorithms [5], which employ the human intervention in the automatic algorithm. The human intervention might be used as pre-process or post-process of the computer algorithms. However, in these algorithms, the human knowledge and computer algorithms were mutually independent. In many cases, either the human pre-process intervention biases the algorithm or the algorithm biases the post-process might be encountered.

In the last decade, the semi-supervised learning algorithms have been much developed so as to address the challenge of the limitation of automatic methods, most of which employed human knowledge in order to guide the automatic algorithms [6]. The key factors of performing an effective semi-supervised method include how the human interventions are employed reliably, and how the knowledge instructs the automatic algorithms. Therefore the semi-supervised method must be designed carefully so as to integrate the human intervention naturally, rather than simply pre-process or post-process. In this article, we proposed a semi-supervised spike sorting framework, wherein the human decisions depended on the visual intermediate results reflecting the distribution of high dimension data. Subsequently, the clustering algorithm was carried out according to the human guideline. In addition, our contribution involves proposing the manifold learning algorithm to describe the structure of SOM grid, whose output is transferred to a proper clustering method referred to as affinity propagation clustering (APC). At the meanwhile, the feature of data reduction of SOM decreases the time-complexity and space-complexity of APC.

The outline of this paper is introduced as follows. We first introduced three components constituting our sorting framework. Next we gave two experiments, in which the procedures and results of the proposed method were illustrated. We concluded this study by discussing the experimental results and the characteristics of our semi-supervised clustering framework.

2 Method

As mentioned above, our semi-supervised spike sorting framework consisted of three components, i.e. self-organizing map [7], the geodesic distance matrix calculated by shortest path algorithm [8], and affinity propagation clustering algorithm [9]. At first, a competitive neural network model called self-organizing map is performed upon the selective high dimension feature space. The output of this phase is only the rough visualization of the feature space, including the clusters' structure and dense information. The operators select a candidate exemplar for each cluster according to this visualization result, followed by performing geodesic distance based affinity propagation clustering that outputs the final classification results. The flowchart of our framework is shown in fig. 1. In the following, these three components will be discussed in detail.

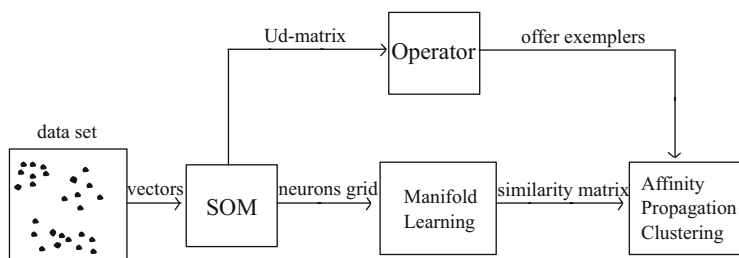


Fig. 1. The flowchart of the proposed semi-supervised spike sorting framework

2.1 Self-Organizing Map

SOM is an unsupervised neural network model, which is a natural visualization and clustering tool. Generally, the output of the SOM is not the final clustering result of original data. In other word, the SOM only outputs the intermediate result describing the cluster structure embedded in the feature space. Although there are literatures proposing post-processing methods for clustering the outputs of SOM, some intrinsic drawbacks still exist, which will be discussed in Section 4. We will also introduce a new clustering algorithm and discuss its efficiency in this study later. The mechanism of the visualization of the SOM will be discussed firstly.

SOM has the feature of preserving the distribution and topology information within the feature space. After adaptive learning from samples, the neurons within the grid are embedded into the feature space, wherein more neurons are arranged in the region with many samples (dense region), and few neurons are arranged in region with few samples (sparse region). We take the distribution of weight vectors after learning as an

input space of SOM, while the original 2-D grid as an output space. Now the input space preserves the cluster structure of feature space. To extract this information into the output space that can be viewed directly by human, some candidate visualization mapping methods are options. Here we will employ Ud-Matrix, the variant of the popular U-Matrix method [10][11][12], to address the problem. The size of Ud-Matrix is equal to the size of output grid. In Ud-Matrix each element represents the dense information of the region around the corresponding neuron. The smaller value the element takes, the denser the region is, and vice versa. If we use the gray color to denote the different values of elements within the Ud-Matrix, the cluster structure within the feature space can be viewed directly.

2.2 Geodesic Distance

Generally, the input space of SOM preserves not only the distribution information, but also the topology structure. It means that the neurons connected in the output space must be the close ones in input space; in contrast, the neurons having long neighboring distance in the output space must be the far ones, even though the Euclidean distance between them may be small. In other words, the input space of SOM has a manifold structure. This suggests that the trained neurons cannot be partitioned simply based on Euclidean distance metric, while it is the common process in the traditional post-processing methods of SOM [12][13].

In order to handle manifold structure in the machine learning framework, in recent years many manifold learning algorithms have been proposed [8]. Among them, a constructive method referred to as ISOMAP [8] employs the concept of geodesic distance instead of Euclidean distance aiming at describing the low dimension manifold more appropriately. The geodesic distance between two samples is defined as the shortest path distance in the k-nearest neighbors connecting network. Here the k-nearest neighbors network means that each sample connects with its k-nearest neighbors so that the shortest path algorithm from graph theory can be carried out upon this connective graph. In case high dimension data having low dimensionality, the geodesic distance outperforms the Euclidean distance no matter what clustering algorithms are performed subsequently [8].

In this study, we employed the geodesic distance metric to describe the topology information preserved within the manifold structure of SOM grid. But the difference between its original version and ours is that we do not need k-nearest neighbors any more, since the SOM grid is a neural network that has been connected itself. Thus the geodesic distance is naturally an appropriate measurement for describing the SOM grid.

2.3 Affinity Propagation

Now the neurons representing the cluster structure of feature space have been outputted from the SOM algorithm, which are waiting to be partitioned. After employing the geodesic distance metric, the information within the SOM grid have been stored in the pair-wise geodesic distance matrix. Thus, it is time to perform the clustering algorithm. Here we choose the affinity propagation clustering algorithm to address the problem due to its efficiency and prominent clustering property. Generally the APC algorithm

works upon a similarity matrix describing the pair-wise similarity within the data set, no matter what similarity metric is. This feature, which is not supported by most other clustering algorithms, enables APC algorithm to handle the geodesic distance matrix (here the similarity is the negative of geodesic distance) offered by manifold learning. The APC algorithm is described briefly in the following.

The APC algorithm takes as input a similarity matrix $s(i, k)$, indicating how well the point with index k is suited to be the exemplar for point i . The algorithm runs iteratively as the messages passing between data points. There are two kinds of messages passed in the procedure so as to generate some ideal exemplars. These two messages are "responsibility" ($r(i, k)$) and "availability" ($a(i, k)$) respectively. After initialization and getting input from similarity matrix, the messages exchange among points are implemented by updating of "responsibility" and "reliability" matrices. After some iteration, the ideal exemplar and the corresponding partition emerges gradually when $a(i, k)$ and $r(i, k)$ matrixes tend to be stable.

The affinity propagation clustering has been proved to work well in some fields [9]; however there are still some drawbacks when it is utilized in practice.

Like k-means algorithm, the first two challenges are how to set the number of clusters and how to choose the better initial exemplars. In our proposed framework, the visualization result generated by SOM is natural prior information for clustering. In our study, the APC algorithm will be guided by human intervention, wherein the initial exemplars are selected by operators depending on the observation of the Ud-Matrix plot. The human knowledge, which is based on the intermediate result calculated by machine, is used to guide the clustering algorithm. Furthermore, the human knowledge is integrated in machine algorithm, rather than is used in pre-process or post-process of machine. That is why our framework is named "semi-supervised".

The second drawback accompanied with APC algorithm is the high space complexity. As we know, the APC algorithm works upon the similarity matrix, whose space complexity is $O(N^2)$. Thus, it is unpractical to calculate the original spike data set directly. Fortunately, the SOM algorithm is also a data compression technique. The size of data could be reduced to the square root scale of the original one after the SOM processing. For example, the data with the size of ten thousand may reduced to the neurons with the size of hundred. Now that the APC algorithm is performed upon the similarity matrix of neurons other than the spikes themselves, obviously the space complexity is linear with the spikes size.

2.4 Overview of the Proposed Framework

Now let's summarize our framework by discussing the role of each component in the framework, which is referred to their contributions and benefits.

- First, the SOM offers the contributions of adaptive learning, data visualization and data reduction, wherein the adaptive learning preserves the distribution and topology information of feature space; data visualization is helpful to human intervention; data reduction enable the efficiency of geodesic distance and APC. Concerning the benefits, due to the fact that the structure underlying the SOM grid is

manifold, the geodesic distance is employed to describe this special structure in the proposed framework.

- Second, other than the contribution of geodesic distance, the computation of this kind of distance has the drawback of space complexity. However, with the assistance of data reduction accompanied with SOM, the drawback has been overcome, which will be discussed further in Section 4.
- Third, most clustering algorithm cannot take as input the similarity matrix. Fortunately the APC has the predominance of handling the similarity matrix, even though the matrix is not symmetrical. Moreover, the APC is the advanced version of k-means and as time efficient as k-means. But the bottleneck of APC is the parameters setting. In the proposed framework, the problem is addressed by human instruction, which is derived from SOM observation.

In summary, the components composing the framework work interdependently, wherein one's disadvantage is compensated by another's advantage. What's more, in this designed framework, the human knowledge is employed reliably to guide the procedure of machine algorithm.

3 Experiments

3.1 Experiment on High SNR Data

This experiment is on a real data recorded from the tetrode recording [14]. In this data set the diversity of recorded signals among different neurons is obvious, wherein the diversity is mainly represented in different electrodes. That is, the amplitude recorded from four electrodes may vary dramatically for various neurons. Although plenty of features extracted from these four electrodes would be the candidates to make up the feature space, we tended to use the first three principal components (which cover more than 80% variances) from each electron, which constitute a 12-dimensional feature space describing this data set.

Fig. 2(a) shows the intermediate result from the SOM. From this Ud-Matrix visualization plot, some latent clusters within the high-dimension feature space are disclosed. This plot consists of ordered neurons that describe the clusters and edges in term of dense information. Dark color denoting short distance means clusters, while light color means edges among clusters. From fig. 2(a) three main clusters (putative neurons) were found (the small cluster in the center was ignored; see Section 4 for the more details). Although this observable partition is a qualitative result, it will be transformed to the quantitative one after APC algorithm. More than an intermediate result, this observable plot is consulted by operator so as to guide the subsequent clustering properly. In this case, the neurons marked with red pluses are chosen to be the most possible centroids of putative clusters. Fig. 2(b) shows the classification result of the SOM neurons partitioned by APC. Note that the results are consistent with the qualitative results showed in fig. 2(a).

After cutting the SOM neurons, the final partition of original spike can be obtained by assigning the original data points to different classes, depending on the class label of the BMU neuron that the point belongs to. Fig. 3 shows the corresponding means

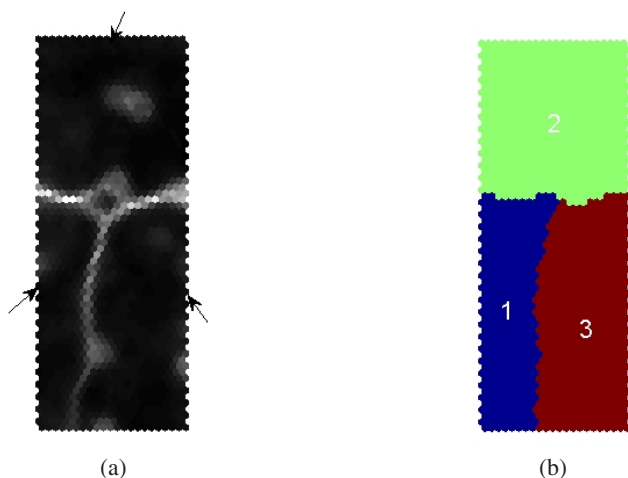


Fig. 2. The gray plot of Ud-matrix and the SOM+APC classification result. (a) denotes the Ud-matrix, arrows point out the candidate centroids for putative clusters chosen by operator, (b) is the neurons partitioned by APC, and the Arabia numbers denote the label of each class.

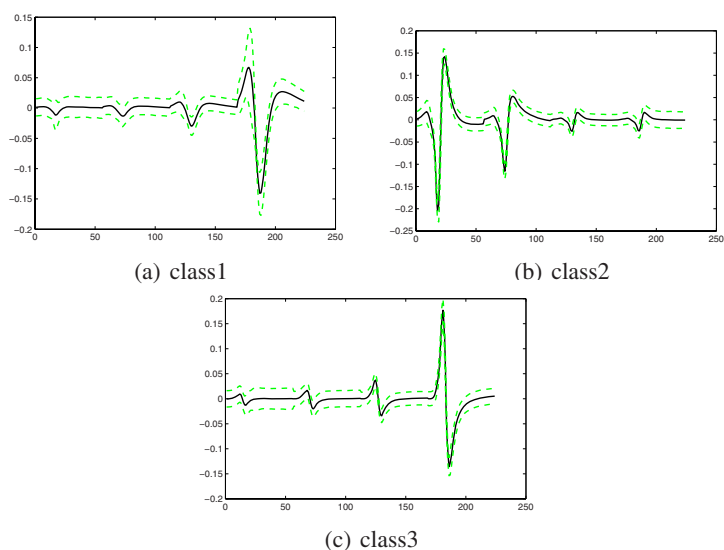


Fig. 3. The means and standard variances of the corresponding classes. Black solid line denotes the mean, green dashed line denotes the value of $\text{mean} \pm \text{standard variance}$, respectively.

and standard variances for each cluster in fig. [2\(b\)](#). Note that the means among various clusters is obviously different, which suggests that each cluster represents different neurons. In addition, the variance of the same mean is small, which indicates that the cluster is so compact that no more clusters can be divided from these clusters.

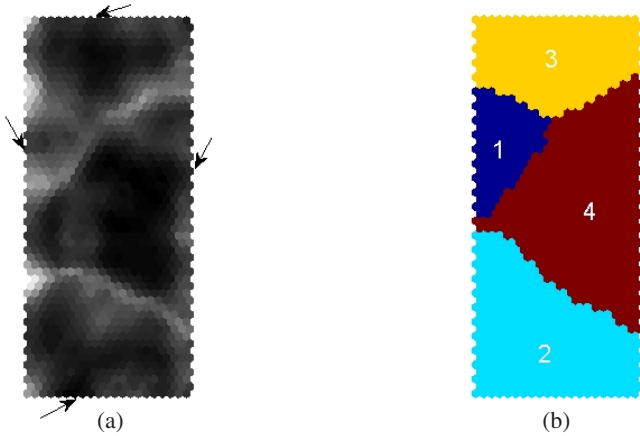


Fig. 4. The gray plot of Ud-matrix and the SOM+APC classification result. (a) denotes the Ud-matrix, arrows point out the candidate centroids for putative clusters chosen by operator, (b) is the neurons partitioned by APC. The Arabia numbers denote class label.

3.2 Experiment on Low SNR Data

This experiment is on another real data recorded from the tetrode recording. Due to the low SNR, the diversity of the recorded signals among different neurons is less obvious than the one in Section 3.1. The main diversity among various clusters is contributed by the different waveforms in the same electrode. In this more difficult case, we will see that different neurons still can be found by our method.

In this experiment, the SNR is lower than the one of the last experiment, so the first 5 principal components from each electron were chosen as feature set so as to cover near 80% variances. These features constitute a 20-dimensional feature space, whose cluster structure was extracted by the SOM and shown in form of gray plot in fig. 4(a). In this visualization plot, different operators may have various decisions on how many latent clusters the plot represents. How to select the reasonable candidate centroids rests with the operator, and whether the decision is correct or not can be reflected by the coming clustering results. Therefore, our method is flexible for human intervention and the performance is observable. Here we tried to divide these SOM neurons into four clusters, whose centroids were marked with pluses in fig. 4(a). The subsequent APC algorithm partitions the neurons into four parts as shown in fig. 4(b). Similar to the last experiment, we can find that the clustering result is consistent with the visualization plot, indicating the reasonability of the APC algorithm.

The same post-processing as the last experiment is performed upon the results of APC; consequently the means and standard variances of the corresponding classes are shown in fig. 5. From these plots, we can see that the classes are obviously different except class 1 and class 3, whose diversity is not obvious. The reason for it may be found in fig. 4 where Class 1 is a small cluster and close to Class 3. In this situation, some operators may tend to combine these two classes. Because the mergence of the final classes is reasonable, either to merger or to separate is laid on the operator's inde-

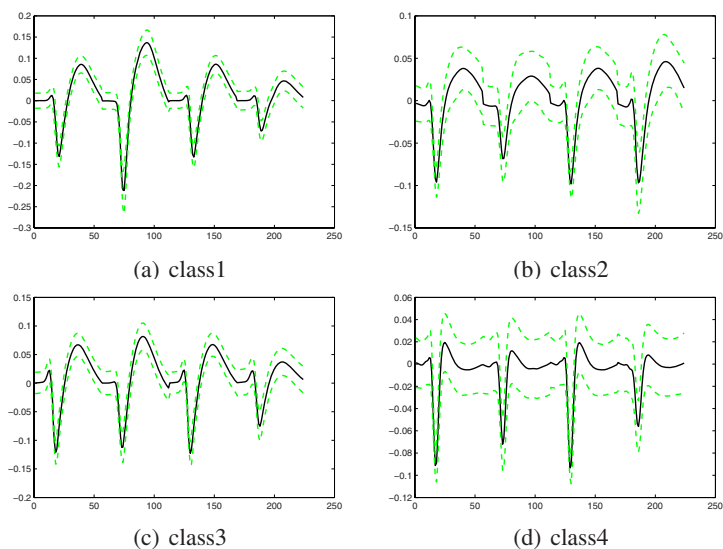


Fig. 5. The means and standard variances of corresponding classes. Solid lines denote the means; dashed lines denote the values of $\text{mean} \pm \text{standard variance}$. The differences between Class 1 and Class 3 are not obvious, so some operators may tend to merger them into one class.

pendently decision for obtaining their optimal result. Thus, based on the flexibility of our framework, the human knowledge can be utilized sufficiently.

4 Discussion

So far, although many methods have been proposed to cluster the neurons of SOM grid, such as the k-means, hierarchical clustering, and the methods imposing the constraint of connection relation, these methods do not reflect the manifold structure underlain by the trained grid. These SOM based post-processing clustering algorithms frequently yield clusters including non-neighboring neurons [12][13]. The SOM grid is a two-dimensionality manifold structure intrinsically, therefore it is obvious that the traditional clustering algorithms can work well only when the manifold learning idea is employed. The basic idea for manifold learning is that the intrinsic structure cannot be measured by Euclidean distance metric, in the case that the high dimension vectors are high correlative due to the intrinsic low dimension structure.

The basic technique for choosing a proper "preference" for APC algorithm is running the algorithm several times initialized by assigning random "preference" for each point, till a best criterion is obtained. However, this non-guide process not only is time consuming but also returns a local optimum frequently. When priori information is available, the setting of parameters would be more reasonable by employing priori human knowledge. What's more, the selection of candidate centroids is robust, which means that the choice of near points as initial exemplar give the same clustering result.

In addition, there are many candidate features for spike sorting. The dense based SOM+APC method can use many possible features, and to choose which subset depends on the operator. Although the features used in our study is the general PCA, more discriminative features in specific case can be chosen. Based on the visualization plot from SOM, the human intervention is considerable and flexible. In order to guide the APC algorithm, the operator not only choose the proper candidate centroids, but also decide a specific cluster either should be ignored or should be considered as a real cluster. For example, in the experiment of Section 3.1, we tended to ignore a very small cluster located in center; while in the experiment of Section 3.2, we tended to consider the cluster 1 as a real class.

5 Conclusions

In this paper, we proposed a semi-supervised spike sorting framework, in which some advanced techniques were constructed to work cooperatively. Since the manual method for spike sorting is limited by invisibility of high-dimension data and time-consuming process, while the automatic methods are unreliable, we managed to make full use of the advantages of both human and machine. Fortunately, the human knowledge and some advanced machine learning techniques are integrated seamlessly in our framework. In the method and experiments section, we introduced our proposed method in details, whose effect has been illustrated to be efficient in the experiments of real data. In addition, we also discussed the reasons why they can work interdependently, and why the human intervention can be employed easily and reasonably. Based on these effective combination and human intervention, our framework performed well.

References

1. Buzsaki, G.: Large-scale recording of neuronal ensembles. *Nature Neuroscience* 7(5), 446–451 (2004)
2. Lewicki, M., et al.: A review of methods for spike sorting: the detection and classification of neural action potentials. *Network: Computation in Neural Systems* 9(4), 53–78 (1998)
3. Quiroga, R., Nadasdy, Z., Ben-Shaul, Y.: Unsupervised Spike Detection and Sorting with Wavelets and Superparamagnetic Clustering (2004)
4. Csicsvari, J., Hirase, H., Czurko, A., Mamiya, A., Buzsaki, G.: Oscillatory Coupling of Hippocampal Pyramidal Cells and Interneurons in the Behaving Rat. *Journal of Neuroscience* 19(1), 274–287 (1999)
5. Harris, K., Henze, D., Csicsvari, J., Hirase, H., Buzsaki, G.: Accuracy of Tetrode Spike Separation as Determined by Simultaneous Intracellular and Extracellular Measurements. *Journal of Neurophysiology* 84(1), 401–414 (2000)
6. Basu, S., Banerjee, A., Mooney, R.: Semi-supervised Clustering by Seeding. In: *Proceedings of 19th International Conference on Machine Learning (ICML 2002)*, pp. 19–26 (2002)
7. Kohonen, T.: The Self-organizing Map. *Proceedings of the IEEE* 78(9), 1464–1480 (1990)
8. Tenenbaum, J., Silva, V., Langford, J.: A Global Geometric Framework for Nonlinear Dimensionality Reduction (2000)
9. Frey, B., Dueck, D.: Clustering by Passing Messages between Data Points. *Science* 315(5814), 972 (2007)

10. Ultsch, A.: Self-organizing neural networks for visualization and classification. In: Proc. Conf. Soc. for Information and Classification (1992)
11. Vesanto, J.: Som-based data visualization methods. *Intelligent Data Analysis* 3, 111–126 (1999)
12. Vesanto, J., Alhoniemi, E.: Clustering of the self-organizing map. *IEEE Transactions on Neural Networks* 11(3), 586–600 (2000)
13. Wen, G., Guo, X., Huang, D., Liu, K.: Application of self-organizing map in aerosol single particles data clustering. In: *Neural Networks, 2007. International Joint Conference on IJCNN 2007*, pp. 991–996 (2007)
14. Lin, L., Chen, G., Xie, K., Zaia, K., Zhang, S., Tsien, J.: Large-scale Neural Ensemble Recording in the Brains of Freely Behaving Mice. *Journal of Neuroscience Methods* 155(1), 28–38 (2006)

A One-Step Network Traffic Prediction

Xiangyang Mu¹, Nan Tang¹, Weixin Gao¹, Lin Li¹, and Yatong Zhou²

¹ School of Electrical Engineering, Xi'an Shiyou University,
Xi'an 710065, China

² School of Information Engineering, Hebei University of Technology,
Tianjin 300401, China
muyou98@xsyu.edu.cn

Abstract. In the information society today computer networks are an indispensable part of people's life. Network traffic prediction is important to network planning, performance evaluation and network management directly. A variety of machine learning models such as artificial neural networks (ANN) and support vector machine (SVM) have been applied in traffic prediction. In this paper, a novel network traffic one-step-ahead prediction technique is proposed based on a state-of-the-art learning model called minimax probability machine (MPM). The predictive performance is tested on traffic data of Ethernet, experimental results show that the predictions of MPM match the actual traffics accurately and the proposed methods can increase the computational efficiency. Furthermore, we compare the MPM-based prediction technique with the SVM-based techniques. The results show that the predictive performance of MPM is competitive with SVM.

Keywords: Network Traffic; Minimax Probability Machine; Support Vector Machine; Prediction.

1 Introduction

Today computer networks are an indispensable part of people's life. network traffic prediction is of significant interest in many domains, including congestion control, admission control and network bandwidth allocation. In high-speed network such as asynchronous transfer mode (ATM), the bandwidth can be allocated based on the accurate traffic prediction, thus ensuring Quality of Service(QoS) of the users and accomplishing the preventive congestion control [1]. Traffic prediction requires accurate modeling techniques which can capture the statistical characteristics of actual traffic [2]. The traditional linear model cannot capture the property of uncertainty and time-variance about network traffic [3]. In order to improve prediction performance, nonlinear models such as artificial neural networks (ANN) [4] and Markov modulated Poisson process (MMPP) models [5] are introduced to capture the real traffic characteristic. However, these models suffer from problems like the existence of local minima or the choice of model structure. Recently, support vector machines (SVM), a new kernel function and modeling technique, has been applied for network traffic prediction [1].

Within the machine learning community, there has been a good deal of excitement about the use of minimax probability machine (MPM) model for regression and prediction in recent years [6][7]. Compared to SVM, MPM pays more attention to the typical rather than the boundary samples [8]. It is in some sense similar to the relevance vector machine proposed in Tipping [9]. Furthermore, the MPM is related to vicinal risk minimization [10], in which SVM were improved using the covariance of the classes to push the hyperplane away from the samples that belong to the class with the largest covariance matrix.

Since the MPM presents many merits, it has been suggested in various applications [11]. However, little attention has been paid to apply MPM to the prediction of network traffic. It is value for us to investigate the problem of whether a good performance could be obtained if we apply MPM to the traffic prediction. In this paper, a novel MPM-based network traffic one-step-ahead prediction technique is proposed and tested on traffic data of Ethernet. Experiments result illustrated that the MPM can capture the uncertainty and time-variance about network traffic. Its predictive performance is comparable to SVM.

The layout of the paper is as follows, we formulate the problem of network traffic prediction in the next section. Following that, a novel MPM model for network traffic prediction is introduced in Sec. 3. Experiments are reported in Sec. 4 and some conclusions are summarized in the end.

2 Network Traffic Prediction

Network traffic presents the number of packets per unit time. The traffic data can be seen as a time series $s(n)$ varied with the time n . We could predict the future traffic level by constructing a prediction model which takes into account the past observations. To be more specific, assume that exists a smooth map $f: R^d \rightarrow R$ such that

$$s(n) = f[s(n-1), s(n-2), \dots, s(n-d)] , \quad (1)$$

If the map f were known, the value of series s at n is uniquely determined by its d values in the past. So the prediction task can be achieved by estimating the map f .

For simplicity of notation, we define the scalar $t_n \equiv s(n)$ and the d -dimensional vector $\mathbf{x}_n \equiv (s(n-1), s(n-2), \dots, s(n-d))^T$ in such a way that Eq.(1) can be written simply as

$$t_n = f(\mathbf{x}_n) . \quad (2)$$

In order to estimate f , a training samples set D_N with capability N can be constructed as follows: $D_N = \{(\mathbf{x}_n, t_n) \in R^d \times R \mid n = 1, \dots, N\}$.

3 Minimax Probability Machine for Network Traffic Prediction

Given the training samples set D_N , the MPM would like to estimate $f(\mathbf{x})$ by finding a model that maximizes the minimum probability of being $\pm \mathcal{E}$ accurate

$$\max \left[\min P \left(\left| f(\mathbf{x}) - t \right| \leq \varepsilon \right) \right] \tag{3}$$

Assume the function $f(\mathbf{x})$ has the form of

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b_r = w^{(1)}x^{(1)} + w^{(2)}x^{(2)} + \dots + w^{(d)}x^{(d)} + b_r \tag{4}$$

Where $\mathbf{w} = (w^{(1)}, w^{(2)}, \dots, w^{(d)})^T$. MPM formulates the Eq. (3) as a binary classification problem to determine the parameters \mathbf{w} and b_r .

The MPM turns each training sample (\mathbf{x}_i, t_i) for $i = 1, \dots, N$ into two class of $d + 1$ dimensional vectors $\tilde{\mathbf{x}}_i$ or $\tilde{\mathbf{y}}_i$. The former is labeled as class $\tilde{\mathbf{x}}$, and the latter one as class $\tilde{\mathbf{y}}$.

$$\tilde{\mathbf{x}}_i = (t_i + \varepsilon, \mathbf{x}_i)^T = (t_i + \varepsilon, x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(d)})^T, \quad i = 1, 2, \dots, N \tag{5}$$

$$\tilde{\mathbf{y}}_i = (t_i - \varepsilon, \mathbf{x}_i)^T = (t_i - \varepsilon, x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(d)})^T, \quad i = 1, 2, \dots, N \tag{6}$$

The above defined artificial classification problem could be solved by any binary classifier. In this paper, we focus on using MPM for classification (MPMC) [8] as the underlying classifier for the problem defined by (5) and (6). Assume the boundary obtained by the MPMC is

$$\mathbf{a}^T \mathbf{u} = b \tag{7}$$

Where

$$\mathbf{u} = (f(\mathbf{x}), \mathbf{x})^T = (f(\mathbf{x}), x^{(1)}, x^{(2)}, \dots, x^{(d)})^T \tag{8}$$

The parameters $\mathbf{a} = (a^{(1)}, a^{(2)}, \dots, a^{(d)}, a^{(d+1)})^T$ and b in Eq. (7) can be determined by following constrained optimization problem

$$\min_{\mathbf{w}} \left(\left\| \mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}^{1/2} \mathbf{a} \right\|_2 + \left\| \mathbf{R}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{1/2} \mathbf{a} \right\|_2 \right) \tag{9}$$

$$s.t. \quad \mathbf{a}^T (\boldsymbol{\mu}_{\tilde{\mathbf{x}}} - \boldsymbol{\mu}_{\tilde{\mathbf{y}}}) = 1$$

where $\boldsymbol{\mu}_{\tilde{\mathbf{x}}}$, $\boldsymbol{\mu}_{\tilde{\mathbf{y}}}$, $\mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}$, and $\mathbf{R}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ satisfy $\tilde{\mathbf{x}} \sim (\boldsymbol{\mu}_{\tilde{\mathbf{x}}}, \mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}})$, and $\tilde{\mathbf{y}} \sim (\boldsymbol{\mu}_{\tilde{\mathbf{y}}}, \mathbf{R}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}})$. In addition, the Ref. [8] gave a kernelized version of the optimization problem (9) by mapping the samples into a high-dimensional feature space.

The boundary (7) obtained by the MPMC turns directly into the prediction function one wants to estimate. That is to say, once the parameters \mathbf{a} and b have been determined; we use the classification boundary to predict the output $f(\mathbf{x})$ for a new input \mathbf{x} . When substituting expression (8) in (7), we obtain

$$a^{(1)}f(\mathbf{x}) + a^{(2)}x^{(1)} + a^{(3)}x^{(2)} + \dots + a^{(d+1)}x^{(d)} = b \tag{10}$$

The Eq. (10) can be reformulated as

$$f(\mathbf{x}) = -\frac{a^{(2)}}{a^{(1)}}x^{(1)} - \frac{a^{(3)}}{a^{(1)}}x^{(2)} - \dots - \frac{a^{(d+1)}}{a^{(1)}}x^{(d)} + \frac{b}{a^{(1)}} \tag{11}$$

Compared the Eq. (11) to Eq. (4), it is derived

$$\begin{aligned} w^{(i)} &= -a^{(i+1)} / a^{(i)}, \quad i=1,2,\dots,d \\ b_r &= b / a^{(1)} \end{aligned} \quad (12)$$

4 Experiments

Here we use the Ethernet network traffic as real traffic series for prediction. And the Ethernet is collected at Bellcore Morristown Research and Engineering Center. The example presents the number of packets per unit time. And they are aggregated at different timescales of 1 and 5 seconds. In this paper we only consider the timescales of 1 second.

Before prediction, two different types of traffic data are normalized to the interval $[0, 1]$. After that, we construct 120 samples for training and 100 for testing candidate models. Mean square error (MSE) and prediction error (PE) are used as performance measures in prediction. Their definitions are

$$MSE = \frac{1}{NT} \sum_{n=1}^{NT} [s(n) - f(\mathbf{x}_n)]^2, \quad PE(n) = s(n) - f(\mathbf{x}_n), \quad (13)$$

where NT is the number of test samples, $s(\cdot)$ is the actual series and $f(\cdot)$ is the prediction.

The kernel functions for MPM and SVM are both based on Gaussian kernel, that is

$$K(\mathbf{u}_n, \mathbf{u}) = \exp\left(-\|\mathbf{u}_n - \mathbf{u}\|^2 / \sigma^2\right) \quad (14)$$

Where the kernel parameter σ is determined by a simple cross-validation technique. For the MPM model, the parameter \mathcal{E} is set for 0.4. For the SVM, the regularization factor is set for 10^3 , and the insensitive loss parameter is set for 0.02. For simplicity, the dimension parameter d is fixed as 3.

Fig.1 show samples of the prediction results in graphical form. The actual values of traffic (at timescale 1 second) are shown as solid line, the corresponding predicted values are superimposed on the actual values as dotted line and dashed line respectively. As it can be seen from the figures, we find the predictions of MPM match the actual traffics accurately.

In the experiment we acquire that the training time of SVM is 26.7 seconds, and it cost only 1.3 seconds for MPM. So the proposed methods can increases the computational efficiency.

To analyze the effect of the parameter d on the predictive performance, we change the value of d from 3 to 5 and 8. The predictive performance is quantified by the MSE and is shown in Table 1. We can see from the table that the performance of MPM and SVM is comparable. The results also show that the parameter d has a significant effect on the performance. In addition, we notice that for each model, using a larger d in prediction usually gives a smaller MSE. However, the prediction speed would reduce significantly when the parameter d is too large.

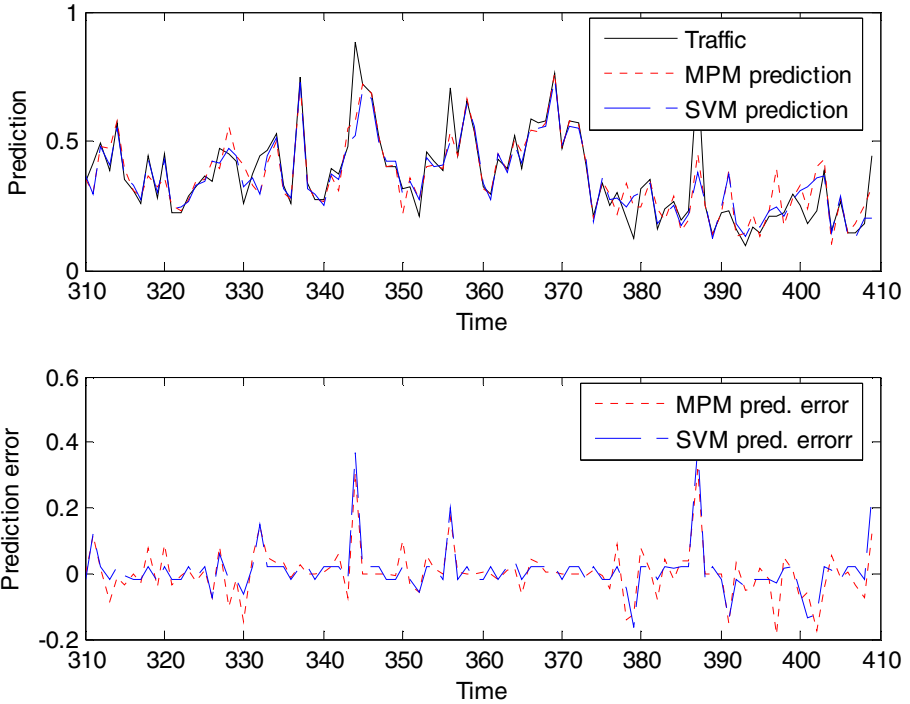


Fig. 1. The actual and predicted values of Ethernet traffic. The upper graph shows traffic (solid line), the prediction using MPM (dotted line) and SVM (dashed line). The lower graph shows the prediction error curves of two models.

Table 1. Prediction performance on traffics expressed in terms of mean square error (MSE)

Models	d	MSE ($\times 10^{-3}$) Ethernet
MPM	3	5.706
	5	0.392
	8	0.412
SVM	3	5.554
	5	0.380
	8	0.386

5 Conclusions

To avoid the drawbacks of traditional learning models for network traffic prediction, a novel MPM-based traffic one-step-ahead prediction technique is proposed in this paper. The prediction performance is tested on traffic data of Ethernet. The experiments demonstrate that the proposed technique attains satisfactory performance in prediction accuracy and the predictive performance of MPM is competitive with

SVM. Therefore, the proposed technique can be used for congestion control in high-speed network, to meet the user QoS requirements.

References

1. Liu, Z.X., Zhang, D., Liao, H.: Multi-scale Combination Prediction Model with Least Square Support Vector Machine for Network Traffic. In: Wang, J., Liao, X.-F., Yi, Z. (eds.) ISNN 2005. LNCS, vol. 3498, pp. 385–390. Springer, Heidelberg (2005)
2. Khotanzad, A., Sadek, N.: Multi-scale High-speed Network Traffic Prediction Using Combination of Neural Networks. In: Proceedings of the International Joint Conference on Neural Networks, vol. 2, pp. 1071–1075 (July 2003)
3. Chen, B.S., Peng, S.C., Wang, K.C.: Traffic Modeling, Prediction, and Congestion Control for High-Speed Networks: A Fuzzy AR Approach. *IEEE Trans. Fuzzy Systems* 8(5), 491–508 (2000)
4. Hall, J., Mars, P.: Limitation of Artificial Neural Network for Traffic Prediction in Broad Networks. *IEEE Proc. Comm.* 147, 114–118 (2000)
5. Sang, A., Li, S.Q.: A Predictability Analysis of Network Traffic. *Computer Networks* 39(4), 329–345 (2002)
6. Strohmman, T.R., Grudic, G.Z.: A Formulation for Minimax Probability Machine Regression. In: Dietterich, T.G., Becker, S., Ghahramani, Z. (eds.) *Advances in Neural Information Processing Systems (NIPS)*, vol. 14. MIT Press, Cambridge (2002)
7. Strohmman, T.R. and Grudic, G.Z.: Robust Minimax Probability Machine Regression. *Journal of Machine Learning Research* (submitted), <http://nago.cs.colorado.edu/~strohman/>
8. Lanckriet, G.R.G., Ghaoui, L.E., Bhattacharyya, C., Jordan, M.I.: A Robust Minimax Approach to Classification. *Journal of Machine Learning Research* 3, 555–582
9. Tipping, M.E.: The Relevance Vector Machine. In: Solla, S., Leen, T.K., Muller, K.-R. (eds.) *Advances in Neural Information Processing Systems (NIPS)*, vol. 12. Morgan Kaufmann, San Francisco (2002)
10. Chapelle, O., Weston, J., Bottou, L., Vapnik, V.: Vicinal Risk Minimization. In: Leen, T. (ed.) *Advances in Neural Information Processing Systems (NIPS)*, vol. 13, pp. 416–422. MIT Press, Cambridge
11. Hoi, C.H., Lyu, M.R.: Robust Face Recognition using Minimax Probability Machine, Tech. Rep., Dep. Computer Science and Engineering, The Chinese University of Hong Kong (2004)

A Two-Step Selective Region Ensemble for Facial Age Estimation

Shenglan Ben, Guangda Su, and Youshou Wu

Department of Electronic Engineering, Tsinghua University, Beijing, China
bs106@mails.tsinghua.edu.cn

Abstract. The effect of aging varies in different facial regions. The significance of regions' age related changes also differs in each age range. In this paper, an efficient subset is selected from all possible rectangle regions in the face image to form a global ensemble on the whole age range. Age range-based selective ensembles are also formed in a similar way. Based on those selective ensembles, a two-step selective region ensemble method is proposed for age estimation. In this framework, the first step is using the global ensemble to give a prediction of possible age range. The second step is to use the ensemble on the predicted age range to make a final estimation. Experiments show that using selective region ensemble can improve age estimation performance, and age range-based selective region ensemble is even superior to the global ensemble.

Keywords: region selection; selective ensemble; facial image; age estimation; age range.

1 Introduction

Human face is a prolific information source conveys significant information including identity, emotion, gender, age, etc. Among these, age is one of the basic attributes crucial to human communication. People at different age differ in aesthetics, linguistics, consumption habit and have different requirements. This makes Age Specific human computer interaction in large demand.

The effect of aging varies in facial regions. Some regions exhibit more identity information while others exhibit more age related information. Moreover, the significance of facial regions' age related changes are different at each age range. Thus, using more significant ageing related regions in different age groups can emphasize aging information and reduce the influence of other variations including the identity difference.

In this paper, we first propose an automatic image region selection algorithm for facial age estimation. Based on this selection algorithm, a two-step framework for facial age estimation is proposed. In this framework, the first step is the coarse age estimation. In this step we ensemble the selected regions in the whole age range to give a coarse estimation of the facial age, this estimation predicts possible age range of that face. The second step is a finer estimation using the ensemble of the significant regions in the predicted age range to give a final estimation of the age.

The rest of this paper is organized as follows. In Section 2, the related work is briefly reviewed. In Section 3, the selection algorithm is introduced. The two-step framework of age estimation is introduced in Section 4. In Section 5 experiments are reported. Finally conclusions are presented in section 5.

2 Related Work

Although having vast potential applications, age estimation problem is still underdeveloped. Apart from a large amount of literatures in aging synthesis and rendering, there are relatively few publications concerning age estimation.

Existing facial age estimation methods can be divided into 3 categories [12]: anthropometric model, aging pattern subspace methods, and aging function methods.

Anthropometric model was first proposed by Kwon and Lobo [6] to handle the facial age estimation problem. Based on geometric ratios and skin wrinkle analysis, they classified input face images into one of the three groups: babies, young adults and senior adults. Also in this category, the research of Horng *et al.*[4] and Hayashi *et al.*[3] proposed approaches using different geometric ratios, wrinkle analysis and classification methods to solve the problem.

The aging function method is the first continues age estimation method. It was proposed by Lanitis[8]. In their work, a polynomial function called aging function is proposed to model the relationship between the parametric description of the face images and the ages: $age = f(b)$, where b is the model parameters and $f(\cdot)$ is the polynomial aging function.

The aging pattern subspace (AGES) model was proposed by Geng [11] to handle highly incomplete data due to the difficulty in data collection. AGES learned a subspace to model the aging pattern, which was defined as a sequence of personal aging face images. The ageing pattern of a test face image was determined by the projection in the subspace that can best reconstruct the face image and the age was indicated by the position of the image in that aging pattern.

Few researches have been done on using different facial part for age estimation. Some similar work was done in face recognition research. Brunelli and Poggio [1] used four masks respectively to get the eyes, nose, mouth and the whole face for face recognition. They claimed that local features could be better than global ones. Pentland *et al.* [9] used eigeneyes, eigennoses and eigenmouths for face recognition. They indicated that the eigenfeatures alone were sufficient in achieving a comparable result to that of eigenface, and the combination of eigenface and eigenfeatures could achieve better performance. Geng and Zhou [10] proposed an automatic region selection method for face recognition method. In their work all possible regions in the face image were regarded as a kind of feature, and based on the error-ambiguity decomposition, their search method was proposed to select regions that were accurate and diverse.

Lanitis [7] performed automatic age estimation experiments using four different facial regions: the whole face (including the hairline), the whole internal face, the upper part of the face and the lower parts of the face. According to his result, the internal part of the face and in particular the area around the eyes proved to be the most significant for the task of automatic age estimation.

3 Face Region Selection for Age Estimation

The significance of age estimation is different in facial regions. Using the facial parts where aging information is emphasized can result in more accurate estimations. Although Lanitis [7] compared the significance of the four facial regions in age estimation, this is a very rough division. The most suitable size of the regions remains unknown.

To avoid the unstable factor in region selection, an automatic algorithm is proposed to select the best combination of regions from all the rectangle regions in the face image.

As mentioned in reference [10], multi-region age estimation methods can also be viewed as special ensemble learning process, i.e. each region can be used to train a weak estimator, and then an ensemble is built from the weak estimators. According to Krogh and Vedelsby [5], the generalization error of an ensemble is $E = \bar{E} - \bar{A}$, where \bar{E} and \bar{A} are the average generalization error and average ambiguity of component estimators respectively. Therefore, to get a strong ensemble, the component estimators should be accurate and diverse.

The number of possible rectangle regions, denoted by L , would be very huge even the image is small. But most of the L regions are either with poor estimation performance or too similar to others. Enlightened by Zhou’s GASEN algorithm [13], we first use the genetic algorithm to select a possible subset from all the rectangle regions in face image. The pre-selection algorithm is shown in Table 1. In this method, weighted averaging is used to ensemble all the regions. The pre-selection problem can be solved by an optimization problem which minimize the generalization of the selective

ensemble of the regions: $w^* = \arg \min_w \left(\sum_{i=1}^L w_i f_i(x_j) - age_j \right)$, where $f_i(\cdot)$ is the estimator of the i th region, w_i is its weight in the ensemble and $\sum_{i=1}^L w_i = 1$. Random weights are first assigned to each region. Then genetic algorithm is used to

Table 1. The Pre-Selection Algorithm

Given : Training set T . Validation set V The rectangle region set of the face image $R = \{R_i, i = 1, 2, \dots, L\}$, An possible subset of N region should be selected from R ,	
<ol style="list-style-type: none"> 1. Train a estimator based on each region R_i ($N = 1, 2, \dots, L$), using the training set T. 2. Generate a population of the regions’ weight vectors 3. Evolve the population where the fitness function of a weighted vector is $g(w) = 1 / \hat{E}_w^V$. 4. w^* = the evolved best weighting vector 5. Choose the N possible regions $\{PR_1, PR_2, \dots, PR_N\}$ corresponding to the N largest weights 	
Output: $PR = \{PR_1, PR_2, \dots, PR_N\}$ is the pre-selected subset.	

evolve those weights. The evolved weights can be regard as a measure of the regions' significance in the ensemble. N regions with the largest weights are selected for further processing.

A validation data set is used to evaluate the goodness of the individuals in the evolving population. Let $\hat{E}_w^V = \sum_{\alpha \in \text{validation}} (f(x_\alpha) - y_\alpha)$ denote the estimated generalization error of the ensemble corresponding to weighting vector w on validation set V , where $f(x) = \sum_{i=1}^L w_i f_i(x)$ is the weighted ensemble of all regions. It's obvious that the smaller \hat{E}_w^V is, the better w is. So $g(w) = 1/\hat{E}_w^V$ is used as the fitness function.

Table 2. The Region Selection Algorithm

Given: Training set T , validation set V the pre-selected possible region subset $PR = \{PR_1, PR_2, \dots, PR_N\}$	
1.	Train a estimator based on each region PR_i ($N = 1, 2, \dots, L$), using the training set T . Use this estimator on all validation images, and get the mean absolute error mae_i
2.	Sort PR_i according to ascending order of mae_i . Get the ordered region sequence: $PR_{i1}, PR_{i2}, \dots, PR_{iN}$.
3.	Set $S = \{PR_{i1}\}$ and $T = \{PR_{i2}, \dots, PR_{iN}\}$, $smae$ = the MAE of region PR_{i1} .
4.	selection procedure: for $t = 1$: Maximun Iteration Number Forward inclusion: 4.1 For each region PR_j in T ,form the ensemble of regions $S \cup PR_j$: $f = \sum_{\alpha \in S \cup PR_j} f_\alpha(x) / S \cup PR_j $. Calculate its MAE on V , denote by t_j . 4.2 Find the largest t_j , denoted by t_l , and its corresponding added region R_l . 4.3 If $t_l \leq smae$, $S = S \cup R_l$, $T = T / R_l$, and $smae = t_l$ Else goto 5 Backward exclusion: 4.4 if $ S \leq 2$, goto 4, else goto 4.5 4.5 For each region PR_j in T , form the ensemble of regions S / PR_j : $f = \sum_{\alpha \in S / PR_j} f_\alpha(x) / S / PR_j $. Calculate its MAE on V , denote by t_j . 4.6 Find the largest t_j , denoted by t_l , and its corresponding region R_l . 4.7 If $t_l \leq smae$, $S = S / R_l$, $T = T \cup R_l$, and $smae = t_l$ Else goto 4
5.	end selection
Output: the selected region are in S , and the ensemble is $f = \sum_{\alpha \in S} f_\alpha(x) / S $	

When the possible region subset PR is selected, we use the algorithm as shown in Table 2 to select m regions to form an efficient ensemble for age estimation. In this procedure, a forward-inclusion and back-exclusion strategy is used. In each iteration, a region that can mostly improve the age estimation performance is selected, and the worst region that would damage the performance is excluded. Finally the ensemble is built from regions in S by averaging the weak estimators.

4 Two-Step Age Estimation Based on Selective Region Ensemble

The aging effect is various in different facial regions. Facial organs would mature at different ages, and their significant age related variation would appear at different ages. For example, postural wrinkles in forehead usually appear early in one's late 20s while gravitational wrinkles around cheekbones usually appear after one's 40s. For each age range, we can use the most significant regions to estimate age. From this view of point, a two-step framework is proposed in this section to use the diversity of organs' aging. In this framework, we first predict an individual's possible age range, and then use the significant face regions in that age range to make an estimation of age.

The most important problem here is how to define the age ranges. Age ranges are used to be defined as exclusive divisions of the whole age range. For example, 20 to 29 can be defined as an age range while 30 to 39 is another. The division is usually subjectively determined, but in fact the aging modes may be very different from 20 to 29. That is, the regions efficient for age 20 may unnecessarily still efficient for 29. Considering this uncertainty, a range is defined for each age a in our method. For age a , its age range is defined as $Range(a) = [a - R, a + R]$, R is its radius here. To select a proper R , both the changeability of aging modes and the prediction error in the first step should be considered. After that, a range based ensemble is then formed using the algorithm in Table 2.

Now, our new framework is consisted of two steps: The first step is possible age range prediction. The region selection algorithm in Table 2 is run on the whole age range, and a global ensemble is built from the selected regions. A possible age \hat{a}_0 is estimated by this global ensemble, and the age range is $Range(\hat{a}_0)$. The second step is using the selected region ensemble in the predicted age range to make a final estimation.

5 Experiments

5.1 Face Database

Our OAPFD (One Age Per-Person Face Database) Age Database is used for the experiments.

The database contains 865 high-resolution face images from 865 different Chinese male subjects, i.e. each subject has only one image. The subjects' age ranges from 23 to 60, most of the ages have 25 examples per age. Since the face images are all frontal images of Asian males, taken indoors, with neutral expressions, the variations in pose,



Fig. 1. Typical images from the OAPFD Age Database

illumination, expression and makeup is limited. Typical samples of the database are shown in Fig 1.

5.2 Methodology

The Appearance Model [2] is used here as a face feature extractor. The shape of a face image is represented by 110 landmark points, and its texture is represented by aligning each image to the mean shape of the training set. An example of the shape and texture of a image is shown in Fig 2. We do PCA to the shapes and textures respectively to get eigenspaces for shape and texture.

Aging function is used in our experiment. Since only one face image exists for each subject, the Global Aging Function [8] is used to indicate the relationship between the face features and the age.

Lanitis [8] empirically selected the quadratic function since it offered the best alternative of the performance and computation. But their experiments were based on FG-NET Aging Database, most of whose ages were ranged from 0 to 35. One problem is whether it's still holds for other age ranges?

We have performed an experimental evaluation to assess the linear and quadratic aging functions. The results are shown in Table 3. The results demonstrate that, quadratic function is not better than the linear ones on our OAPFD database with the age range from 23 to 60. For its simplicity, the linear aging function will be used in our experiments.

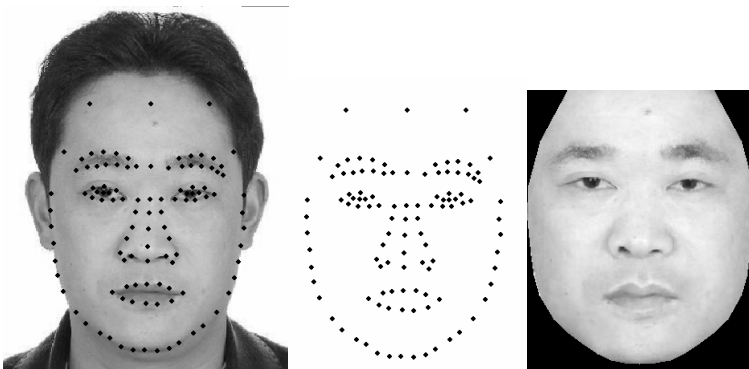


Fig. 2. Shape and Texture of a Face Image (From right to left: face image, shape image, texture image)

Table 3. Comparison between the Linear and Quadric Function

	Linear Function	Quadric Function
Mean Absolute Error (years)	5.23	5.44
Standard Deviation (years)	0.3716	0.3712

We use mean absolute error, abbreviating as MAE, to measure the average performance of the age estimators. Also a cumulative score at error level a [11] is used to measure the accuracy at each error level, it's calculated by:

$$CumScore(a) = M_{e \leq a} / M \times 100\% \quad (1)$$

Where $M_{e \leq a}$ is the number of test images with absolute error no higher than a , M is the number of test images. The cumulative score is an indicator of the age estimators' accuracy at error level a . Bigger cumulative scores at lower error levels indicates better age estimation performance.

For convenience of comparison, we defined a random estimation error. Without loss of generality, for every image in the test set, the random estimation of its age is assumed as 40. So the corresponding random estimation error is $RandomError = \sum_{\alpha \in \text{test set}} |40 - age_{\alpha}| = 9.6$ in our OAPFD Database.

5.3 The Effect of Shapes in Age Estimation

Intuitively, age related shape variation would be subtle in our OAPFD Database since all the subjects are adults whose skeleton growth is matured. To confirm this assumption, only the shape information is used to estimate age. Its MAE is 12.43, which is even worse than random estimation. It indicates that, shapes here exhibits more identity information rather than age information.

In the following experiments, only the shape normalized textures will be used to avoid the negative effects of personalized face shapes.

5.4 Two-Step Framework for Age Estimation

In the training step of this experiment, 15 images of each age are randomly chosen as training set and the others are used as validation set. In the test step, the validation set is added into the training set, while the same number of image is chosen as test set from the original training set. We perform 10 runs and their results are averaged for region selection. The parameter N of Algorithm 1 is set to 350 in this experiment.

Global Ensemble of Selected Regions

In this experiment, the region selection algorithm in table 2 is used to get a global ensemble of regions on the whole age range, and the ensemble is then used on test set to estimate ages. The first 4 selected regions are shown in Fig 3. The age estimation results are shown in Table 4 and Fig 5. The MAE and cumulative score of the global ensemble are both superior than Lanitis' global linear aging function method using

internal face region. It can be seen from Fig 3 that the internal face (the face region from brow to mouth) is the most efficient region for age estimation, and age-sensitive local regions: the region near the middle axis, the eye region, etc. are also selected. The result is in good agreement with our intuition.

Age Range-based Selective Region Ensemble

We firstly verified the intuition that in different age ranges the most significantly changed regions are different. Aging changes facial appearance, on the other hand, a particular face appearance indicates its corresponding age. From this point of view, the age estimation ability can reflect the region's significance of age-related variation. The regions have best age estimate performance at age 25,35,45 and 55 are listed in Fig 4. From this figure we can see, when getting older, more localized regions act. This might because facial aging is not significant in younger ages, when getting older, more folds and wrinkles appear in local regions, such as eye regions and nasolabial fold. So, local regions become more efficient in older ages. Also, we can see that the best region at age 45 is similar to the second region in global ensemble, this may partly explain the reason why region ensemble works: multiple regions efficient at different ages are add to the ensemble to compromise in the final estimation.

Two-step Age estimation

In this experiment, the two-step age estimation framework is used: the global ensemble is used to give a prediction of age range, and then the ensemble on that age range is used to give the final estimation. The results are compared in Table 4, and their cumulative score are shown in Fig 5. The internal face region here is the face region from upper brow to lower lips, which has the best age estimation performance in all regions. A global aging function is trained on that region and used as a baseline for comparison. Both MAE and cumulative score reveals that using selective region ensemble can improve age estimation performance, and age range-based selective region ensemble is superior than the global ensemble. It also gets a higher cumulative score at lower absolute error levels.

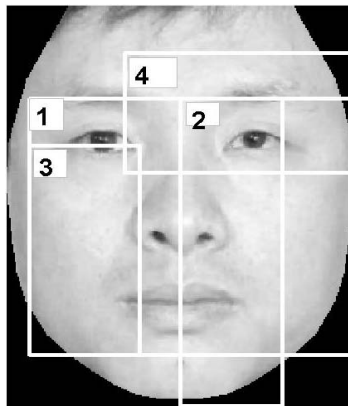


Fig. 3. First Four Regions Selected in Global Ensemble

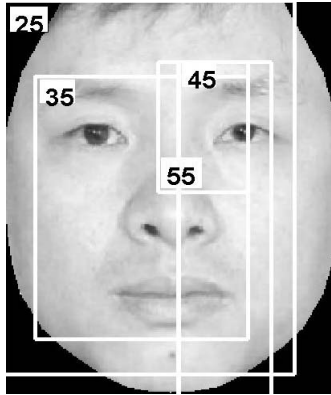


Fig. 4. Regions with best age estimation performance at age 25,35,45 and 55

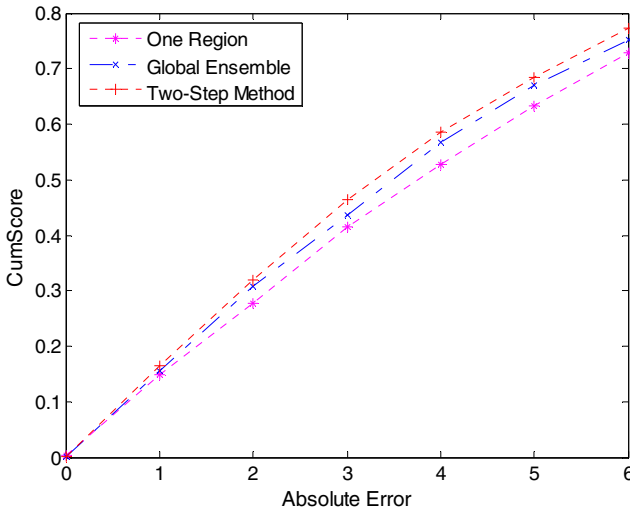


Fig. 5. Cumulative Score of the Age estimation Methods

Table 4. MAE of Internal Face Region, Global Ensemble and The Two-Step Method

	Internal Face Region	Global Ensemble	The two-step Method
MAE	4.354	4.065	3.949

6 Conclusion

In this paper, a two-step selective region ensemble method is proposed for age estimation. Different facial regions contain different individual information. Some regions reveal more age information while others may exhibit more identity information. In different age ranges, the significance of age related variation is also different in

regions. From this point of view, our two-step method is proposed. The first step is to select an efficient subset from all possible rectangle regions in the face image, and then this subset is used to train a global ensemble on the whole age range. The global ensemble is used to give a prediction of a face's possible age range. The second step is to use the ensemble on the predicted age range to make a final estimation.

Using our OAPFD Age Database, we experimentally confirmed our intuition of the diversity of aging modes in different facial regions. And the age estimation results also confirmed the efficiencies of selective region ensemble for age estimation in both global and age range-based situations.

In this paper, since only one image per person exists, global aging function is used to model aging. It is interesting to note that, our age estimation results on OAPFD, even only the internal face region is used, is comparable to methods using multiple aged images per person. For example, AGES [11] gets a MAE at 6.77 and weighted person specific aging function [8] gets a MAE at 4.33, and both of them are experimented on FG-NET Age Database. This may be because images in FG-NET Database have a lower quality and contain more other variations such as illumination, expression and pose. One image per person facial age database is easier to get and its quality can be higher: we can take photos of volunteers at different ages under constrained conditions. This perhaps indicates that one image per person facial age database can be used for efficient continuous age estimation. This will be further investigated in the future work.

References

1. Brunelli, R., Poggio, T.: Face Recognition: Feature versus Templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 15(10), 1042–1052 (1993)
2. Edwards, G.J., Lanitis, A., Cootes, C.J.: Statistical Face Models: Improving Specificity. *Image Vision Comput.* 16(3), 203–211 (1998)
3. Hayashi, J., Yasumoto, M., Ito, H., Koshimizu, H.: Age and Gender Estimation Based on Wrinkle Texture and Color of Facial Images. In: 16th International Conference on Pattern Recognition, pp. 405–408. IEEE Press, Quebec (2002)
4. Horng, W.B., Lee, C.P., Chen, C.W.: Classification of Age Groups Based on Facial Features. *Tamkang Journal of Science and Engineering* 4(3), 183–192 (2001)
5. Krogh, A., Vedelsby, J.: Neural Network Ensembles, Cross Validation, and Active Learning. In: Tesauro, G., Touretzky, D., Leen, T. (eds.) *Advances in Neural Information Processing Systems*, vol. 7, pp. 233–238. MIT Press, Cambridge (1995)
6. Kwon, Y.H., Lobo, N.V.: Age Classification from Facial Images. *Computer Vision and Image Understanding* 74(1), 1–21 (1999)
7. Lanitis, A.: On the Significance of Different Facial Parts for Automatic Age Estimation. In: 14th International Conference on Digital Signal Processing, vol. 2, pp. 1027–1030. IEEE Press, Los Alamitos (2002)
8. Lanitis, A., Taylor, C.J., Cootes, T.F.: Toward Automatic Simulation of Aging Effects on Face Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24(4), 442–455 (2002)
9. Pentland, A., Moghaddam, B., Starner, T.: View-based and Modular Eigenspaces for Face Recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 84–91. IEEE Press, Seattle (1994)

10. Xin, G., Zhou, Z.H.: Image Region Selection and Ensemble for Face Recognition. *J. Comput. Sci. & Technol.* 21(1), 116–125 (2006)
11. Xin, G., Zhou, Z.H., Smith-Miles, K.: Automatic Age Estimation Based on Facial Aging Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29(12), 2234–2240 (2007)
12. Yun, F., Ye, X., Huang, T.S.: Estimating Human Age by Manifold Analysis of Face Pictures and Regression on Aging Features. In: 2007 IEEE International Conference on Multimedia and Expo., pp. 1383–1386. IEEE Press, Los Alamitos (2007)
13. Zhou, Z.H., Wu, J.X., Jiang, Y., Chen, S.F.: Genetic Algorithm Based Selective Neural Network Ensemble. In: Proceedings of the 17th International Joint Conference on Artificial Intelligence, Seattle, WA, vol. 2, pp. 797–802 (2001)

A Unified String Kernel for Biology Sequence

Dehui Yuan, Shengyun Yang, and Guoming Lai

Department of Mathematics and Information Technology,
Hanshan Normal University, Guangdong Province, China, 521041
ydh1x1,@hstc.edu.cn

Abstract. In this paper, we introduce a unified String Kernel. Based on this unified string kernel, we construct improved sparse kernel and composite kernel. Using the same target families and the same test and training set splits as in the protein classification experiments from Weston, we do experiments with these new kernels. The results show that our kernels are superior to previously developed string kernel.

Keywords: unified kernel; improved kernel string; composite kernel string.

1 Introduction

Breakthroughs in large-scale sequencing have led to a surge in the available protein sequence information that has far out-stripped our ability to experimentally characterize their functions. As a result, the analysis of protein sequences becomes one of the most successful areas in bioinformatics, and researchers are increasingly relying on computational techniques to classify these sequences into functional and structural families based on sequence homology. Three major ideas and intuitions are used over and over again in many of the successful techniques. These include the use of local alignments, the use of homologous sequences and the use to secondary structure information.

In the past few years, more and more Kernels appropriate for sequence objects (or string) were suggested, see in [1,2,3,4,5,6,7,8,9,10,11]. These methods represent a sequence X as a vector in feature space and differ on the scheme they employ to actually determine if a particular dimension u (i.e. kmer) is present (i.e. has a non-zero weight) in X 's vector or not. The Spectrum kernel [1] considers u to be present if X contains u as a substring. The Mismatch kernel [2,3], which was shown to match the performance of the Fisher kernel [4] in remote homology detection, considers u to be present if X contains a substring that differs with u in at most a predefined number of positions (i.e. mismatches). And the profile kernel [5] considers u to be present if X contains a substring whose PSSM-based ungapped alignment score with u is above a user-supplied threshold.

For classification of protein sequence data with SVM, several novel families of string kernels, which are similar to the mismatches kernel, were presented. These kernels include restricted gappy kernels, substitution kernels, wildcard kernels, and LA-Kernel method [12], homology kernel [13]. The homology kernel, which

takes into account intuitions from local alignment, sequence homologues and predicted secondary structure, demonstrates how the homology kernel can be used for protein family classification and outperforms state-of-the-art methods for remote homology detection. However, this kernel is sensitive to the parameter.

By analyzing these kernels presented in literatures, we proposed a general framework for constructing string kernels. The rest of the paper is organized as follows. Section 2 introduces a general framework for constructing string kernels. Section 3 discusses the improved sparse kernel. Section 4 describes some composite kernels. Section 5 describes our experimental settings and results, and finally we provide conclusions and future research directions in Section 6

2 A General Framework for Constructing String Kernel

The main idea behind the embedding defined by string kernels such as the mismatch kernel, the homology kernel, the restricted gappy kernel, the substitution kernel, the wildcard kernel, is that the sequences are represented as a collection of its fixed length substrings (of length k) obtained by defining the neighbourhood of k -mers. These string kernels have their roots in the spectrum kernel [9] where the technique was first applied to biological sequences. These fixed length substrings are mapped into a feature space which consists of a dimension for each possible fixed length substring.

Since these kernels only count the occurrences of k -mers or mismatch k -mers in sequences, and a considerable loss of information may result from this restriction. Then, Thomas Lingner and Peter Meinicke gave a new feature space for representation of protein sequences, which is based on histograms for counting distances between oligomers. For more details, see [11].

Combining these two ideas, we introduce a general framework for constructing string kernel. Let Σ be a finite set which we call the alphabet. The elements of Σ are characters. Any $x \in \Sigma^k$ ($k = 0, 1, 2 \dots$) is called a string. Σ^+ represents the set of all non empty strings defined over the alphabet ($\Sigma^+ = \bigcup_{k>0} \Sigma^k$, where Σ^k is the set of sequences of length k). In the following we will use $u, v, \alpha, \beta, w, x, y \in \Sigma^+$ to denote strings and $a, b, c \in \Sigma$ to denote characters. $|x|$ denotes the length of x , $xy \in \Sigma^+$ the concatenation of two strings x and y , ax the concatenation of a character and a string. Moreover, we write $\Sigma' = \Sigma \cup \{*\}$, where $*$ and Σ'^+ are the consensus letter and the set of strings over Σ' respectively. Sequence x contains k -mer α (denoted by $\alpha \in x$, also called a k -gram in the literature), if $x = u\alpha v$. In this case we can write $\alpha = x^p$ if a , the first letter of α , is at the p -th position of x . Let I be an index set which is a finite set constructed over Σ^+ or Σ'^+ , for example, $I = \Sigma^k$ or $\Sigma^k \times \Sigma^k$. And let $Nb(\alpha)$ be a subset of I defined through α . Moreover, we use w to denote the element of index set I , ϕ to denote the mapping:

$$\alpha \rightarrow \phi(\alpha) = (\phi_{w_1}(\alpha), \dots, \phi_{w_n}(\alpha)) \in R^n,$$

where $n = |I|$, $\phi_{w_i}(\alpha) = \text{Weight}(w_i)$ if $w_i \in Nb(\alpha)$ and 0 otherwise, $\text{Weight}(w_i)$ is the weight of w_i in the index set I . For a string $s \in A^+$, $|s| > |\alpha|$, $\phi(s)$ is defined as follows:

$$\phi(s) = \sum_{\alpha \in s} \phi(\alpha)$$

and the general kernel for strings s_1 and s_2 is present as follows:

$$\text{General}(s_1, s_2) = \langle \phi(s_1), \phi(s_2) \rangle \tag{1}$$

Notation: 1) let $I = \Sigma^k$ and $Nb(\alpha) = \{\alpha\}$, we get the spectrum kernel [1];

2) if $Nb(\alpha) = N_{(k,m)}(\alpha) = \{\beta | k\text{-length sequence } \beta \text{ differ from } \alpha \text{ at most } m \text{ mismatches}\}$ and $I = \Sigma^k$, then we get the mismatch string kernel [2];

3) let $I = \Sigma^k$ and $Nb(\alpha)$ is the profile neighborhood $PN_{(k,\sigma)}(\alpha)$, a set of k -length strings which differ from α with a log-probability not greater than σ , than we have the profile kernel [8].

4) let x_{ir} denote the r -th homologue of sequence $x_i (1 \leq r \leq t)$, let x_i^p denote the substring of length k centered at position p in sequence x_i and x_{ir}^p denote the substring of length k of the r -th aligned sequence. Taking $Nb(x_i^p) = \{x_{ir}^p | 1 \leq r \leq t\}$ and $I = \Sigma^k$, we have the local homology kernel [13].

5) let $I = (\Sigma')^k$, notes that $\Sigma^k \subset (\Sigma')^k$, we can use the set Σ^k to define the set $Nb(\alpha)$ for k -mer $\alpha \in \Sigma^k$. For a given k -mer α and $\tau \in (0, 1)$, we define $\text{Match}(\alpha, w_i)$, $\text{Weight}(w_i)$ and ϕ according to index set $(\Sigma')^k$ as follows:

$$\text{Match}(\alpha, w_i) = \begin{cases} 1, & \text{if } w \in \Sigma'^k \text{ matches } \alpha \in \Sigma^k \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

$$\text{Weight}(w_i) = \sqrt{\tau^l (1 - \tau)^{k-l}}, l \text{ is the number of } * \text{ in } k\text{-mer } w_i \tag{3}$$

$$\phi : \alpha \rightarrow \phi(\alpha) = (\phi_{w_1}(\alpha), \dots, \phi_{w_n}(\alpha)) \in R^n, \tag{4}$$

where $n = |(\Sigma')^k|$, $\phi_{w_i}(\alpha) = \text{Match}(\alpha, w_i) \cdot \text{Weight}(w_i) (i = 1, 2, \dots, n)$, and consensus letter “*” matches with every character $c \in \Sigma'$. A k -mer $x \in \Sigma'^k$ with a wildcard * at position p will match to k -mer that differ from it only at position p , for $p = 1, \dots, |x|$. Therefore, we get a weighted kernel of the general kernel defined in (1). And the weighted kernel is also the sparse kernel defined in [13].

6) let $D = \{1, 2, \dots, N\}$ and $I = \Sigma^k \times \Sigma^k \times D$. For a given k -mer $\alpha = x^p$, denote $Nb(\alpha) = \{(x^p, x^{p+d}, d) | 1 \leq d \leq N\}$, then $Nb(\alpha) \subset \Sigma^k \times \Sigma^k \times D$. Applying this new feature mapping ϕ to the general kernel (1), the oligomer kernel [11] can be deduced.

7) for $x \in \Sigma^+$, let $D = \{1, 2, \dots, |x|\}$, $I = \Sigma^k \times D$. For a given k -mer $\alpha = x^p$, denote $Nb(\alpha) = \{(\alpha, q) | \text{if } k\text{-mer } \alpha \text{ appears at least } q \text{ times in } x, q \in D\}$. Then the normalized spectrum kernel [13] can be obtained according to (1).

3 Improved Sparse Kernel

In this section, we present an improved sparse kernel, which is not sensitive to the parameter. In this kernel, we not only punish the mismatching but also highlight the matching. The kernel is presented as follows:

$$ISK(\alpha, \beta) = \frac{\tau^{d_H(\alpha, \beta)} \cdot (1 + \tau - \tau^2)^{k - d_H(\alpha, \beta)}}{(1 - \tau)^{k - d_H(\alpha, \beta)}} \tag{5}$$

where $d_H(\alpha, \beta)$ is the Hamming distance or number of mismatches between k -mers α and β , τ is a parameter and $\tau \in (0, 1)$. Note that Equation (5) can be presented as follows:

$$ISK(\alpha, \beta) = \frac{(1+\tau-\tau^2)^k}{(1-\tau)^k} \cdot \frac{\tau^{d_H(\alpha, \beta)}(1-\tau)^{d_H(\alpha, \beta)}}{(1+\tau-\tau^2)^{d_H(\alpha, \beta)}} = \left(\frac{(1+\tau-\tau^2)}{(1-\tau)} \right)^k \cdot \frac{(\tau(1-\tau))^{d_H(\alpha, \beta)}}{(1+\tau-\tau^2)^{d_H(\alpha, \beta)}}$$

we can expect the improved sparse kernel is not sensitive to the parameter such as the sparse kernel [13]. Furthermore, our experiments in Section 5 show this property. Therefore, we need not readjust the parameter when use this method to new data set.

In order to implement the improved sparse kernel, we only need replace the weight of k -mer w_i $Weight(w_i)$ in Equation (3) by the following weight:

$$Weight(w_i) = \sqrt{\frac{\tau^l}{(1-\tau)^{k-l}}}, l \text{ is the number of } * \text{ in } k\text{-mer } w_i \tag{6}$$

where l is the number of wild cards in the w and $\tau \in (0, 1)$. This improved sparse kernel also can be derived through the general kernel (1), and the method is similar to the Notation 5).

Proposition 1. *The mapping in Equation (4) with the weight in Equation (6) satisfies the property of Equation (5)*

Proof. Let α and β be two k -mers with Hamming distance $d_H(\alpha, \beta) = d$. Then the dot product $\langle \phi(x), \phi(y) \rangle = \sum \phi_{w,q}(x) \cdot \phi_{w,q}(y)$ is the sum $\phi_w^2(\alpha)$ summed over all w such that w matches α and β . Such k -mers w must contain wild cards on the mismatches (there are d such positions) and either the consensus letter or a wild card at all other locations (the rest $k - d$ positions). Therefore,

$$\begin{aligned} \langle \phi(x), \phi(y) \rangle &= \sum_{i=0}^{k-d} \binom{k-d}{i} \left(\frac{\tau^{d+i}}{(1-\tau)^{k-d-i}} \right) \\ &= \frac{\tau^d}{(1-\tau)^{k-d}} \sum_{i=0}^{k-d} \binom{k-d}{i} (\tau(1-\tau))^i 1^{k-d-i} \\ &= \frac{\tau^d}{(1-\tau)^{k-d}} [1 + \tau(1-\tau)]^{k-d} = \frac{\tau^d(1+\tau-\tau^2)^{k-d}}{(1-\tau)^{k-d}} \quad \square \end{aligned}$$

Above we described the improved sparse kernel for a single string of length k . For a longer sequence, we can simply add the images of its substrings together. We refer to this embedding as the unnormalized improved sparse kernel and can be computed by:

$$UISK(x_i, x_j) = \sum_{x_i^p \in x_i, x_j^q \in x_j} \frac{\tau^{d_H(x_i^p, x_j^q)} \cdot (1+\tau-\tau^2)^{k-d_H(x_i^p, x_j^q)}}{(1-\tau)^{k-d_H(x_i^p, x_j^q)}}. \tag{7}$$

Alternatively, if we use the same index set as in Notation 7), that is $\phi_{w,q}(x)$ indexed both by the string of length $w \in \Sigma'^k$ and an integer $1 \leq q \leq |x| - k$ with the matching and Weight function:

$$\text{Match}_{w,q}(\alpha) = \begin{cases} 1 & \text{if } k\text{-mer } w \text{ matches at least } q \text{ substrings in } x \\ 0 & \text{otherwise.} \end{cases}$$

$$\text{Weight}(w, q) = \sqrt{\frac{\tau^l}{(1-\tau)^{(k-l)}}, l \text{ is the number of } * \text{ in } k\text{-mer } w_i$$

and define the $\phi(\alpha)$

$$\phi(\alpha) = (\phi_{w_1,1}(\alpha), \dots, \phi_{w_1,q}(\alpha), \phi_{w_2,1}(\alpha), \dots, \phi_{w_2,q}(\alpha), \phi_{w_n,1}(\alpha), \dots, \phi_{w_n,q}(\alpha))$$

where $\phi_{w_i,j}(\alpha) = \text{Match}_{w_i,j}(\alpha) \cdot \text{Weight}(w_i, j)$, then we get the normalized improved kernel which is similar to the normalized sparse kernel [13].

4 Composite Kernel for String

Comparing the definition of different kernels in section 2, we know that the Sparse kernel contains the information of mismatching kernel and mismatching kernel contains the information of spectral kernel. Take $d = 0$, we get the spectral kernel from oligomer kernel. To realize the oligomer kernel, it is only instructive to consider monomer ($k = 1$), dimer counts ($k = 2$) and trimer counts ($k = 3$), the k is smaller than the one of mismatching kernel where the k is often 5. Obviously, it is impossible to contain the information of mismatching kernel in oligomer kernel. Of great interest, we consider parameterized combinations of two kernels which include oligomer kernel and mismatching kernel. Given the oligomer kernel k_{oli} and mismatching kernel k_{mis} , we consider the convex combination or direct sum kernel [15]:

$$k_{oli+mis} = \tau k_{oli} + (1 - \tau)k_{mis}$$

where $0 < \tau < 1$.

Alternatively, we consider the other two composite approach about oligomer kernel k_{oli} and mismatching kernel k_{mis} . The first one is tensor product kernel of the oligomer kernel k_{oli} and mismatching kernel k_{mis} :

$$k_{oli \cdot mis} = k_{oli} \cdot k_{mis},$$

And the other one is

$$k_{oli \cdot mis}^{log} = \tau \log_2(k_{oli} + 1) + (1 - \tau) \log_2(k_{mis} + 1),$$

where $0 < \tau < 1$.

5 Experiment Results

We use the same 54 target families and the same test and training set splits as in the protein classification experiments from Weston et al [18] in our experiments. Here, each superfamily is represented by positive training and test

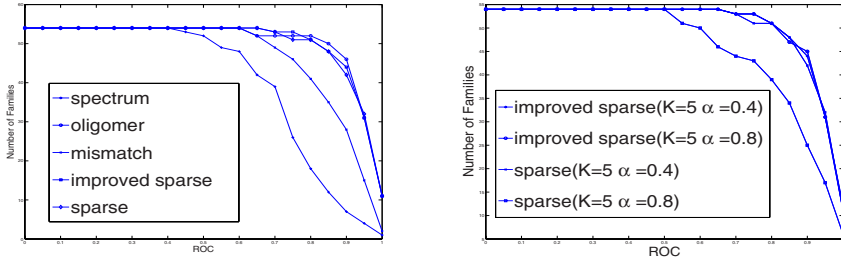


Fig. 1. ROC curves over 54 target families for experiments with different string methods

proteins from families inside the superfamily and by negative training and test proteins from families in other superfamilies. Thereby the number of negative data is much larger than that of the positive ones. In order to cope with the highly unbalanced training sets, we used the SVM entirely with the same parameters from Weston et al [18]. All algorithms are evaluated using receiver operating characteristic (ROC) analysis presented in [17]. An ROC curve plots the rate of true positives as a function of the rate of false positives at varying decision thresholds. The ROC score is the area under this curve. A perfect classifier, which places all positive examples above all negative examples, receives an ROC score of 1, and a random classifier receives a score of 0.5. In addition to the ROC score, we compute the ROC50 score, which is the ROC score computed only up to the first 50 false positives. This score focuses on the top of the ranking, which in some applications is the most important. To our experiment results, the ROC curves are presented in Fig. 5, while ROC50 curves are in Fig. 2. In the same time, Mean ROC50 and ROC scores over 54 target families for experiments with different string methods are summarized in Table 1.

From the results represented in Table 1, we know that the sparse method is sensitive to parameter τ , but ROC of improved sparse method is almost same when parameter τ changes from 0.2 to 0.9. Moreover, we find an interesting thing,

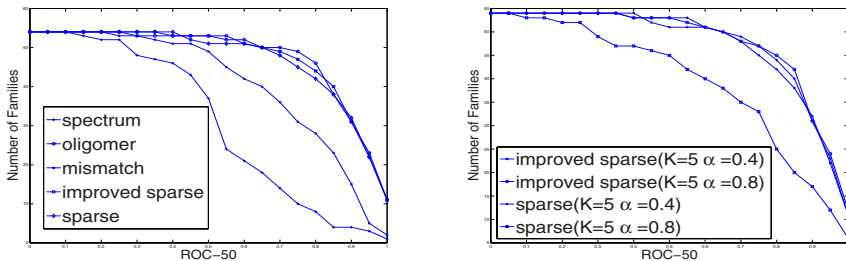


Fig. 2. ROC50 curves over 54 target families for experiments with different string methods

Table 1. Mean ROC50 and ROC scores over 54 target families for experiments with different string methods

Method	ROC	ROC50
sparse($k = 5, \tau = 0.2$)	0.9415	0.8849
sparse($k=5, \tau=0.4$)	0.9384	0.8785
sparse($k = 5, \tau = 0.5$)	0.9261	0.8588
sparse($k = 5, \tau = 0.6$)	0.9011	0.8176
sparse($k= 5, \tau = 0.8$)	0.8539	0.7307
improved-sparse ($k = 5, \tau = 0.2$)	0.9419	0.8867
improved-sparse ($k = 5, \tau = 0.4$)	0.9415	0.8851
improved-sparse ($k = 5, \tau = 0.5$)	0.9414	0.8848
improved-sparse ($k = 5, \tau = 0.6$)	0.9416	0.8852
improved-sparse ($k = 5, \tau = 0.8$)	0.9419	0.8867
oligomer($k=1$)	0.9394	0.8858
mismatch($k = 5, mis = 1$)	0.8713	0.7600
oligomer($k=1,0.4$)+mismatch($k=5,mis=1,\tau=0.6$)	0.9433	0.8887
oligomer($k=1,0.3$)+mismatch($k=5,mis=1,\tau=0.7$)	0.9438	0.8896
oligomer($k=1,0.2$)+mismatch($k=5,mis=1,\tau=0.8$)	0.9453	0.8929
oligomer($k=1,0.1$)+mismatch($k=5,mis=1,\tau=0.9$)	0.9471	0.8970
oligomer($k=1,0.9$)+mismatch($k=5,mis=1,\tau=0.1$)	0.9401	0.8859
oligomer($k=1,0.1 * \log_2$)+mismatch($k=5,mis=1,0.9 * \log_2$)	0.9465	0.8963
oligomer($k=1,0.2 * \log_2$)+mismatch($k=5,mis=1,0.8 * \log_2$)	0.9462	0.8949
oligomer($k=1,0.3 * \log_2$)+mismatch($k=5,mis=1,0.7 * \log_2$)	0.9447	0.8916
oligomer($k=1,0.4 * \log_2$)+mismatch($k=5,mis=1,0.6 * \log_2$)	0.9432	0.8887
oligomer($k=1,0.9 * \log_2$)+mismatch($k=5,mis=1,0.1 * \log_2$)	0.9392	0.8847

that is higher ROC and ROC50 values in $k_{oli+mis}$ and $k_{oli \cdot mis}^{log}$ experiments are results which combination parameter τ corresponded to mismatch kernel is larger than $(1 - \tau)$, although performance of oligomer is better than mismatch kernel generally in experiments.

6 Conclusion and Future Work

On one hand, a general framework for constructing kernel was firstly developed to deal with strings or sequences object. And we proposed the improved sparse kernel with this general kernel. The experiments results show that this new kernel is not sensitive to the parameter τ . It is different from the sparse kernel which is sensitive to parameter α , see the Table 1 or [13]. However, we don't know the reason that this new kernel is not sensitive to the parameter τ . And we will further analyze from theory and experiment results to explain why this modification made the result not sensitive to τ .

On the other hand, the experiments show that the parameterized combinations of two kernels which include oligomer kernel and mismatching kernel can present good results when dealing with string. Motivated by these combinational

kernels, we may propose some new neighborhoods $Nb(\alpha)$ or Index set I which contains the information of oligomer kernel and mismatching kernel at the same time. Therefore, we can use these neighborhoods or index sets to the proposed general kernel in Section 2 to get other new kernels, and expect to get good results. The following two index sets may be useful when dealing with string.

1) If $D = \{1, 2, \dots, N\}$, $I = \Sigma^k \times \Sigma^k \times D$. For a given k -mer x^p , define $Nb(x^p) = \{(\alpha, \beta, d) | \alpha \in N_{k,m}(x^p), \beta \in N_{k,m}(x^{p+d}), 1 \leq d \leq N\}$, then $Nb(x^p) \subset \Sigma^k \times \Sigma^k \times D$. Using this new feature mapping ϕ to the general kernel (II), we get a new kernel which is the generalization of the oligomer kernel (III).

2) If $D = \{1, 2, \dots, N\}$, $I = \Sigma^{k_1} \times \Sigma^{k_2} \times D$ and $k_1 < k_2$. For a given k_1 -mer x^p , define $Nb(x^p) = \{(x^p, \beta, d) | \beta \in N_{k_2,m}(x^{p+d}), x^{p+d} \text{ is } k_2\text{-mer}, 1 \leq d \leq N\}$, then $Nb(x^p) \subset \Sigma^{k_1} \times \Sigma^{k_2} \times D$.

References

1. Leslie, C., Eskin, E., Noble, W.S.: The Spectrum Kernel: A String Kernel for SVM Protein Classification. In: Proceedings of the Pacific Symposium on Biocomputing (PSB), Kaua'i, Hawaii (2002)
2. Leslie, C., Eskin, E., Noble, W.S.: Mismatch String Kernels for SVM Protein Classification. *Adv. Neural Inf. Process. Syst.* 20, 467–476 (2003)
3. Leslie, C., Eskin, E., Cohen, A., Weston, J., Noble, W.S.: Mismatch String Kernels for Discriminative Protein Classification. *Bioinformatics* 20, 467–476 (2004)
4. Jaakkola, T., Diekhans, M., Haussler, D.: A Discriminative Framework for Detecting Remote Protein Homologies. *J. Comput. Biol.* 7, 95–114 (2000)
5. Kuang, R., Ie, E., Wang, K., Wang, K., Siddiqi, M., Freund, Y., Leslie, C.: Profile-based String Kernels for Remote Homology Detection and Motif extraction. *J. Bioinform. Comput. Biol.* 3, 527–550 (2005)
6. Vishwanathan, S., Smola, A.: Fast Kernels for String and Tree Matching. *Adv. Neural Inf. Process. Syst.* (2002)
7. Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., Watkins, C.: Text Classification Using String Kernels. *Journal of Machine Learning Research* 2, 419–444 (2002)
8. Rangwala, H., Karypis, G.: Profile-based Direct Kernels for Remote Homology Detection and Fold Recognition. *Bioinformatics* 21, 4239–4247 (2005)
9. Leslie, C., Kuang, R.: Fast String Kernels Using Inexact Matching for Protein Sequences. *Journal of Machine Learning Research* 5, 1435–1455 (2004)
10. Vinokourov, A., Soklakov, A.N., Saunders, C.: A Probabilistic Framework for Mismatch and Profile String Kernels. In: Proceedings of the 13th European Symposium on Artificial Neural Networks, pp. 325–330 (2005)
11. Lingner, T., Meinicke, P.: Remote Homology Detection Based on Oligomer Distances. *Bioinformatics* 22, 2224–2231 (2006)
12. Saigo, H., Vert, J.P., Ueda, N., Akutsu, T.: Protein Homology Detection Using String Alignment Kernels. *Bioinformatics* 20, 1682–1689 (2004)
13. Eskin, E., Snir, S.: The Homology Kernel: A Biologically Motivated Sequence Embedding into Euclidean Space. In: Proceedings of the 2005 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, pp. 179–186 (2005)
14. Ben-Hur, A., Noble, W.S.: Kernel Methods for Predicting Protein Cprotein Interactions. *Bioinformatics* 21, i38–i46 (2005)

15. Mak, B., Kwok, J.T., Ho, S.: A Study of Various Composite Kernels for Kernel Eigenvoice Speaker Adaptation. In: Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Montreal, Canada, vol. 1, pp. 325–328 (May 2004)
16. Diego, I.M., Moguerza, J.M., Munoz, A.: Combining Kernel Information for Support Vector Classification. In: Roli, F., Kittler, J., Windeatt, T. (eds.) MCS 2004. LNCS, vol. 3077, pp. 102–111. Springer, Heidelberg (2004)
17. Hanley, J.A., McNeil, B.J.: The Meaning and Use of the Area under a Receiver Operating Characteristic (ROC) curve. *Radiology* 143, 29–36 (1982)
18. Weston, J., Leslie, C., Ie, E., Zhou, D., Elisseeff, A., Noble, W.S.: Semi-supervised Protein Classification Using Cluster Kernels. *Bioinformatics* 21, 3241–3247 (2005)

An Adaptively Constructing Multilayer Feedforward Neural Networks Using Hermite Polynomials*

L. Ma¹ and K. Khorasani²

¹ Department of Applied Computer Science, Tokyo Polytechnic University, 1583 Iiyama, Atsugi, Kanagawa, Japan 243-0297

maly@cs.t-kougei.ac.jp

² Department of Electrical and Computer Engineering, Concordia University, Montreal, Quebec, Canada H3G 1M8

kash@ece.concordia.ca

Abstract. In this paper a new strategy is introduced for constructing a multi-hidden-layer feedforward neural network (FNN) where each hidden unit employs a polynomial function for its activation function that is different from other units. The proposed scheme incorporates a structure level adaptation as well as a function level adaptation methodologies in constructing the desired network. The activation functions considered consist of orthonormal Hermite polynomials. Using this strategy, a FNN can be constructed as having as many hidden layers and hidden units as dictated by the complexity of the problem being considered.

Keywords: Adaptive structure neural networks; Hermite polynomials; Multilayer feedforward neural networks.

1 Introduction

The design of a feedforward neural network (FNN) requires consideration of three important issues. The solutions to these will significantly influence the overall performance of the neural networks (NN) as far as the following two considerations are concerned: (i) recognition rate to new patterns, and (ii) generalization performance to new data sets that have not been presented during network training.

The first problem is the selection of data/patterns for network training. This is a problem that has practical implications and has received limited attention. The training data set selection can have considerable effects on the performance of the trained network. The second problem is the selection of an appropriate and efficient training algorithm from a large number of possible training algorithms that have been developed in the literature. Many new training algorithms

* This research was supported in part by the NSERC (Natural Science and Engineering Research Council of Canada).

with faster convergence properties and less computational requirements are being developed by researchers in the NN community. The third problem is the determination of the network size. This problem is more important from a practical point of view when compared to the above two problems, and is generally more difficult to solve. The problem here is to find a network structure as small as possible to meet certain desired performance specifications. What is usually done in practice is that the developer trains a number of networks with different sizes, and then the smallest network that can fulfill all or most of the required performance requirements is selected. This amounts to a tedious process of trial and errors.

The 2nd and 3rd problems are actually closely related to one another in the sense that different training algorithms are suitable for different NN topologies. Therefore, the above three considerations are indeed critical when a NN is to be applied to a real-life problem. This paper focuses on developing a systematic procedure for an automatic determination and/or adaptation of the network architecture for a FNN.

Algorithms that can determine an appropriate network architecture automatically according to the complexity of the underlying function embedded in the data set are very cost-efficient, and thus highly desirable. Efforts toward the network size determination have been made in the literature for many years, and many techniques have been developed [3], [6] (and the references therein).

2 Hermite Polynomials

In this section, the Hermite polynomials with their hierarchical nonlinearities are first introduced. These polynomials will be used subsequently as the activation functions of the hidden units of the proposed constructive feedforward neural network (FNN). The orthogonal Hermite polynomials are defined over the interval $(-\infty, \infty)$ of the input space and are given formally as follows [11] (see also the references therein):

$$H_0(x) = 1, \tag{1}$$

$$H_1(x) = 2x, \tag{2}$$

⋮

$$H_n(x) = 2xH_{n-1}(x) - 2(n - 1)H_{n-2}(x), \quad n \geq 2. \tag{3}$$

The definition of $H_n(x)$ may be given alternatively by

$$H_n(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n} \left(e^{-x^2} \right), \quad n > 0, \quad \text{with } H_0(x) = 1. \tag{4}$$

The polynomials given in (1)–(3) are orthogonal to each other but not orthonormal. The orthonormal Hermite polynomials may then be defined according to

$$h_n(x) = \alpha_n H_n(x) \phi(x), \tag{5}$$

where

$$\alpha_n = (n!)^{-1/2} \pi^{1/4} 2^{-(n-1)/2}, \quad (6)$$

$$\phi(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}. \quad (7)$$

The first-order derivative of $h_n(x)$ can be easily obtained by virtue of the recursive nature of the polynomials defined in (3), that is,

$$\frac{dh_n(x)}{dx} = (2n)^{1/2} h_{n-1}(x) - x h_n(x), \quad n \geq 1, \quad (8)$$

$$\frac{dh_0(x)}{dx} = \alpha_0 \frac{d\phi(x)}{dx} = -x h_0(x), \quad n = 0. \quad (9)$$

In [11], the orthonormal Hermite polynomials are used as basis functions to model 1-D signals in the biomedical field for the purposes of signal analysis and detection. In [4], a selected combination of the orthonormal Hermite polynomials in a weighted-sum form is used as the *fixed* activation function of *all* the hidden units of the OHL-FNN. Hermite coefficients in [8], [9] are used as pre-processing filters, or served as the features of the process, which are then fed to (fuzzy) neural networks for classification problems. A feedforward neural network is designed in [10] that uses Hermite function regression formula to approximate the hidden units activation functions to obtain an improved generalization capability. In this FNN, a fixed number of Hermite polynomials is used.

It was indicated earlier that the Hermite polynomials are chosen due to their suitable properties. There are other polynomials that have similar properties, such as, Legendre polynomials, Tchebycheff polynomials and Laguerre polynomials [5]. However, the input range for these polynomials does not meet the requirement for a hidden activation function which should take values ranging from $-\infty$ to $+\infty$. If one uses a polynomial with a limited input range as an activation function for a hidden unit, one should then restrict the input-side weight space in which an optimal input-side weight vector has to be searched for under a given training criterion. The restriction of the input-side weight space is not desirable as it would limit the representational capability of the network. The justification and rationale for choosing the Hermite polynomials are therefore further motivated by their restriction-free input range characteristic.

Alternatively, the above polynomials may still be considered as activation functions of the hidden units provided that the input to a hidden unit is normalized each time its input-side weight vector is updated. Although, the weight vector is now unconstrained, and only the input is normalized by a properly selected constant, this constant is actually a function of the weight vector. Therefore, the input to the hidden unit during the input-side training phase will be no longer linear with respect to the weight vector. Consequently, as far as the weight training is concerned the input-side training will now have two types of nonlinearities: one arising from the input to the hidden unit, and the other due to the activation function of the hidden unit. The corresponding optimization problem is clearly more complicated as compared to the one with only the nonlinearity of

the activation function. It is, therefore, our conclusion that Hermite polynomials are expected to be the most suitable choice for the activation functions of the hidden units of a FNN.

3 The Proposed Incremental Constructive Training Algorithm

Consider a typical constructive one-hidden-layer feedforward neural network (OHL-FNN) with a linear output layer and a polynomial-type hidden layer, as shown in Figure 1. The output layer can also be nonlinear, as will be shown in our simulation results. The hidden unit with input and output connections denoted by dotted lines is the present neuron that is being trained before it is allowed to join the other existing hidden units in the network. Suppose, without loss of generality, that a given regression problem has an M -dimensional input vector and a scalar one dimensional output. The j -th input and output samples are denoted by $\mathbf{X}^j = (x_0^j, x_1^j, \dots, x_M^j)$ ($x_0^j = 1$ denoting the bias) and d^j (target), respectively, where $j = 1, 2, \dots, P$ (P is the number of training data samples). The OHL constructive algorithm starts from a null network without any hidden units. At any given point during the constructive learning process, suppose there are $n - 1$ hidden units in the hidden layer, and the n -th hidden unit is being trained before it is added to the existing network (see Figure 1). The output of the n -th hidden unit for the j -th training sample is given by

$$f_n(s_n^j) = h_{n-1} \left(\sum_{m=0}^M w_{n,m} x_m^j \right) \tag{10}$$

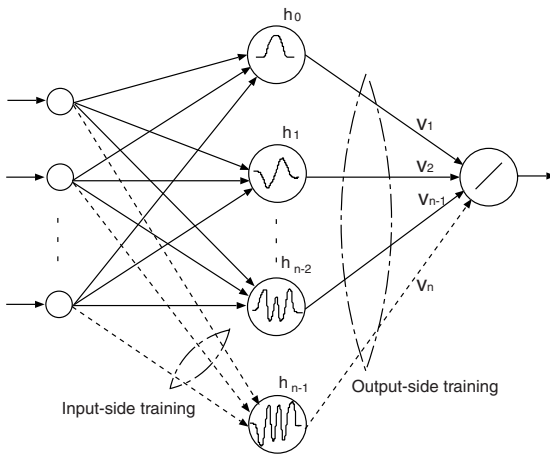


Fig. 1. Structure of a constructive OHL-FNN that utilizes the orthonormal Hermite polynomials as its activation functions for the hidden units

where $h_n(\cdot)$ denotes the n -th order Hermite orthonormal polynomial. Its derivative can be calculated recursively from equation (8) and s_n^j is the input to the n -th hidden unit, given by

$$s_n^j = \sum_{m=0}^M w_{n,m} x_m^j \tag{11}$$

where $w_{n,0}$ is the bias weight of the node with its input $x_0^j = 1$, and $w_{n,m}$ ($m \neq 0$) is the weight from the m -th component of the input vector to the n -th hidden node.

The output of the network may now be expressed as follows

$$\begin{aligned} y_{n-1}^j &= v_0 + \sum_{i=1}^{n-1} v_i h_{i-1}(s_i^j) \\ &= v_0 + v_1 h_0(s_1^j) + v_2 h_1(s_2^j) + v_3 h_2(s_3^j) + \dots + v_n h_{n-1}(s_{n-1}^j). \end{aligned} \tag{12}$$

where v_0 is the bias of the output node, and v_1, v_2, \dots, v_n are the weights from the hidden layer to the output node.

Clearly, the above expression has a very close resemblance to a functional series expansion that utilizes the orthonormal Hermite polynomials as its basis functions, and where each additional term in the expansion contributes to improving the accuracy of the function that is being approximated. Through this series expansion, the approximation error will become smaller as more terms, having higher-order nonlinearities, are included. This situation is actually quite similar to that of an incrementally constructing OHL-FNN, in the sense that the error is expected to become smaller as new hidden units having hierarchically higher-order nonlinearities are successively added to the network. In fact, it has been shown that under certain circumstances incorporating new hidden units in a OHL-FNN can decrease monotonically the training error [7]. Therefore, in this sense adding a new hidden unit to the network is somewhat equivalent to the addition of a higher-order term in a Hermite polynomial-based series expansion. It is in this sense that the present network is envisaged to be more suitable for constructive learning paradigm as compared to a fixed-structure FNNs. Note that, only if $s_1^j = s_2^j = \dots = s_{n-1}^j$, then y_{n-1}^j will be an exact Hermite polynomial-based series expansion. Otherwise, as structured in the algorithm above the expansion would be only approximate since the weights associated with each added neuron is adjusted separately, resulting in different inputs to the activation functions.

The problem addressed now is to determine how to train the n -th hidden unit that is to be added to the active network. There are many objective functions (see for example [2], [6] for more details) that can be considered for the input-side training of this hidden unit. A simple, but a general cost function that has been shown in the literature to work quite well, is as follows [2]:

$$J_{input} = \left| \sum_{j=1}^P (e_{n-1}^j - \bar{e}_{n-1})(f_n(s_n^j) - \bar{f}_n) \right| \tag{13}$$

where

$$f_n(s_n^j) = h_{n-1}(s_n^j) \tag{14}$$

$$\bar{e}_{n-1} = \frac{1}{P} \sum_{j=1}^P e_{n-1}^j, \tag{15}$$

$$\bar{f}_n = \frac{1}{P} \sum_{j=1}^P h_{n-1}(s_n^j), \tag{16}$$

$$e_{n-1}^j = d^j - y_{n-1}^j \tag{17}$$

where $f_n(s_n^j)$ (or $h_{n-1}(s_n^j)$) is an orthonormal Hermite polynomial of order $n - 1$ used as the activation function of the n -th hidden unit, e_{n-1}^j is the network output error when the network has $n - 1$ hidden units and d^j is the j -th target output to be used for the network training.

The derivative of J_{input} with respect to the weight $w_{n,i}$ is calculated by

$$\begin{aligned} \frac{\partial J_{input}}{\partial w_{n,i}} &= sgn \left(\sum_{j=1}^P (e_{n-1}^j - \bar{e}_{n-1})(f_n(s_n^j) - \bar{f}_n) \right) \\ &\times \sum_{j=1}^P (e_{n-1}^j - \bar{e}_{n-1}) h'_{n-1}(s_n^j) x_i^j \end{aligned} \tag{18}$$

where $sgn(\cdot)$ is a sign function. The first-order derivative $h'_{n-1}(s_n^j)$ in the above expression can be easily evaluated by using the recursive expression (8) for $n \geq 2$ and (9) for $n = 1$.

A candidate unit that maximizes the objective function (13) will be incorporated into the network as the n -th hidden unit. Following this stage, the output-side training is performed by solving a least squared (LS) problem given that the output layer has a linear activation function (see Figure 1). Specifically, once the input-side training is accomplished, the network output y^j with n hidden units may now be expressed as follows:

$$y^j = \sum_{k=0}^n v_k f_k(s_k^j), \quad j = 1, 2, \dots, P \tag{19}$$

where v_k , ($k = 1, 2, \dots, n$) are output-side weights of the k -th hidden unit, and v_0 is the bias of the output unit with its input being fixed to $f_0 = 1$. The corresponding output neuron tracking error is now given by

$$\begin{aligned} e^j &= d^j - y^j \\ &= d^j - \sum_{k=0}^n v_k f_k(s_k^j), \quad j = 1, 2, \dots, P. \end{aligned} \tag{20}$$

Subsequently, the output-side training is performed by solving the following *LS* problem given that the output layer has linear activation function, that is

$$J_{output} = \frac{1}{2} \sum_{j=1}^P (e^j)^2 = \frac{1}{2} \sum_{j=1}^P \left\{ d^j - \sum_{k=0}^n v_k f_k(s_k^j) \right\}^2 .$$

After performing the output-side training, a new error signal e^j is calculated for the next cycle of input-side and output-side training. Our proposed constructive FNN algorithm may now be summarized according to the following steps:

Step 1: Initialization of the network

Start the network training process with a OHL-FNN having one hidden unit.

Set $n = 1$, $e^j = d^j$, and the activation function $= h_0(\cdot)$.

Step 2: Input-side training for the n -th hidden unit

Train only the input-side weights associated with the n -th hidden unit $h_{n-1}(\cdot)$. The input-side weights for the existing hidden units, if any, are all frozen. One candidate for the n -th hidden unit with random initial weights is trained using the objective function defined in (13), based on the “quick-prop” algorithm [11]. If instead, for the n -th unit, a pool of candidates are trained, then the one candidate that yields the maximum objective function will be chosen as the n -th hidden unit to be added to the active network.

Step 3: Output-side training

Train the output-side weights of all the hidden units in the present network.

When the output layer has a linear activation function the output-side training may be performed by, for example, invoking the pseudo-inverse operation resulting from the least square optimization criterion as indicated earlier.

When the activation function of output layer is nonlinear, the Quasi-Newton algorithm is to be used for training.

Step 4: Network performance evaluation and training control

Evaluate the network performance by monitoring the metric known as fraction of variance unexplained (FVU) [7] metric on the training data set. The FVU measure is defined according to

$$FVU = \frac{\sum_{j=1}^P (\hat{g}(\mathbf{x}^j) - g(\mathbf{x}^j))^2}{\sum_{j=1}^P (g(\mathbf{x}^j) - \bar{g})^2} \tag{21}$$

where $g(\cdot)$ is the function being implemented by the FNN, $\hat{g}(\cdot)$ is an estimate of $g(\cdot)$ or the output of the trained network, and \bar{g} is the mean value of $g(\cdot)$. The network training is terminated provided that certain stopping conditions are satisfied. These conditions may be specified formally, for example, in terms of a prespecified FVU threshold or a maximum number of permissible hidden units, among others. If the stopping conditions are not satisfied, then the network output y^j and the network output error $e^j = d^j - y^j$ are computed, n is increased by 1, i.e., $n = n + 1$, and we proceed to **Step 2**.

It is well-known in the constructive OHL-FNNs literature [7] that the determination of a proper initial network size is a problem that needs careful attention. This initialization may significantly influence the effectiveness and efficiency of the network training that follows. However, in our proposed scheme, the network training starts from the smallest possible architecture (that is the “null” network). This is an important advantage of our proposed algorithm over the similar methods previously presented in the literature.

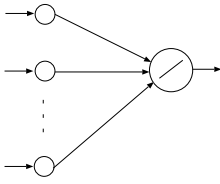
4 Proposed Multi-hidden-layer FNN Strategy

Construction and design of multi-hidden-layer networks are generally more difficult than those of one-hidden-layer (OHL) networks since one has to determine not only the depth of the network in terms of the number of hidden layers, but also the number of neurons in each layer. In this section, a new strategy for constructing a multi-hidden-layer FNN with regular connections is proposed. The new constructive algorithm has the following features and advantages:

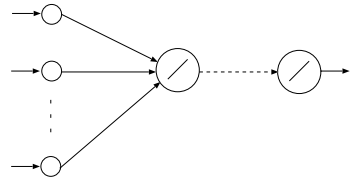
- (1) As with OHL constructive algorithms, the number of hidden units in a given hidden layer is automatically determined by the algorithm. In addition, the number of hidden layers are also determined automatically by the algorithm.
- (3) The algorithm adds hidden units and hidden layers one at a time, when and if they are needed. The input-side weights for all the hidden units in the installed hidden layer are fixed (input weights are frozen), however, the input-side weights for a new hidden unit that is being added to the network are updated during the input-side training. Generally, a constructive OHL may need to grow a large number of hidden units, or even could fail to represent a map that actually requires more than a single hidden layer. Therefore, it is expected that our proposed algorithm that incrementally expands its hidden layers and units automatically as a function of the complexity of the problem, would converge very fast and is computationally more efficient when compared to other constructive OHL networks.
- (4) Incentives are also introduced in the proposed algorithm to control and monitor the hidden layer creation process such that a network with fewer layers are constructed. This will encourage the algorithm to solve problems using smaller networks, reducing the costs of network training and implementation.

4.1 Construction, Design and Training of the Proposed Algorithm

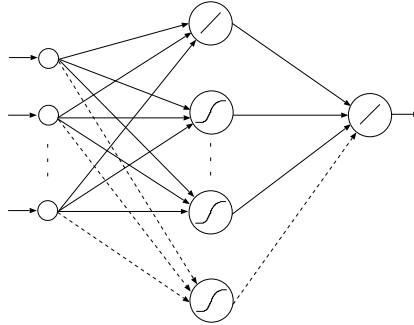
For sake of simplicity, and without loss of any generality, suppose one desires to construct a FNN to solve a mapping or a regression problem having a multi-dimensional input and a single scalar dimensional output. Extension to the case of multi-dimensional output space is quite trivial and straightforward. It is also assumed that the output unit has a linear activation function for dealing with the regression problems, although nonlinear activation function have to be used



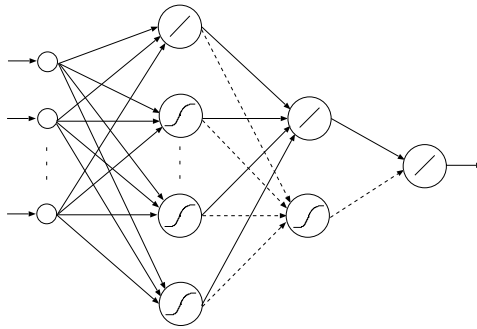
(a) Initial linear network



(b) Creation of the first hidden layer



(c) Addition of hidden units to the first hidden layer



(d) Creation of the second hidden layer and its units

Fig. 2. Multi-hidden-layer network construction process for the proposed strategy using Hermite polynomials for the nonlinear activation function of the neurons

for pattern recognition problems. Specifically, the output-side training of the weights would have to be performed by utilizing the Delta rule, instead of an LS solution that is used here when the activation function is selected as a linear function. The proposed constructive algorithm consists of the following steps:

Step 1: Initially train the network with a single output node and no hidden layers (refer to Figure 2(a)). The training of the weights is achieved according to a LS solution or some gradient-based algorithm, including the bias of the output node. The training error FVU (as described below) is monitored and if it is not smaller than some prespecified required threshold, one would then proceed to Step 2a, otherwise one would go to Step 2c. The performance of a network is measured by the fraction of variance unexplained (FVU) [7] which is defined as

$$FVU = \frac{\sum_{j=1}^P (\hat{g}(x^j) - g(x^j))^2}{\sum_{j=1}^P (g(x^j) - \bar{g})^2} \quad (22)$$

where $g(\cdot)$ is the function to be implemented by the FNN, $\hat{g}(\cdot)$ is an estimate of $g(\cdot)$ realized by the network, and \bar{g} is the mean value of $g(\cdot)$. The FVU is equivalently a ratio of the error variance to the variance of the function being analyzed by the network. Generally, the larger the variance of the function, the more difficult it would be to do the regression analysis. Therefore, the FVU may be viewed as a measure normalized by the “complexity” of the function. Note that the FVU is proportional to the mean square error (MSE). Furthermore, the function under study is likely to be contaminated by an additive noise ϵ . In this case the signal-to-noise ratio (SNR) is defined by

$$SNR = 10 \log_{10} \left\{ \frac{\sum_{j=1}^P (g(x^j) - \bar{g})^2}{\sum_{j=1}^P (\epsilon^j - \bar{\epsilon})^2} \right\} \quad (23)$$

where $\bar{\epsilon}$ is the mean value of the additive noise, which is usually assumed to be zero.

Step 2a: Treat the output node as a hidden unit with unit output-side weight and zero bias (refer to Figure 2(b)). The hidden unit input-side weights are fixed and are as determined from the previous training step. The output error of the new linear output node will be identical to its hidden unit. Proceed to Step 2b.

Step 2b: A new hidden unit is added to the present layer in Step 2a one at a time, just as in a constructive OHL-FNN (refer to Figure 2(c)). Each time a new hidden unit is added, the output error (FVU) is monitored to see if it is smaller than a certain prescribed threshold value. If so, one then proceeds to Step 2c. Otherwise, a new hidden unit is added until the output error FVU is stabilized above the prescribed threshold. Once the error is stabilized, Step 2a is invoked again and a new hidden layer is introduced to possibly further reduce the output error (refer to Figure 2(d)). Steps 2a and 2b are

repeated as many times as necessary until the output error FVU is below the prescribed desired threshold, following which one then proceeds to Step 2c. Input-side training of a new hidden unit weights is performed by maximizing a correlation-based objective function, whereas the output-side training of the weights is performed through a LS type algorithm.

Step 2c: The constructive training is terminated.

For implementing the above algorithm some further specifications are needed. First, a metric “stop-ratio” is introduced to determine if the training error FVU is stabilized as a function of the added hidden unit, namely

$$\text{stop-ratio} = \frac{FVU(n-1) - FVU(n)}{FVU(n-1)} \times 100\% \quad (24)$$

where $FVU(n)$ denotes the training error FVU when the n -th hidden unit is added to a hidden layer of the network. Note that according to [6] addition of a proper hidden unit to an existing network should reduce the training error. The issue that needs addressing here is the magnitude selection of the stop-ratio. At the early stages of the constructive training, this ratio may be large. However, as more hidden units are added to the network the ratio will monotonically decrease. When it has become “sufficiently” small, say a few percents, the inclusion of further additional hidden units will contribute little to the reduction of the training error, and as a matter of fact may actually start giving rise to an increase in the generalization error, as the unnecessary hidden units may force the network to begin to memorize the data. Towards this end, when the stop-ratio is determined to be smaller than a prespecified desired percentage *successively* for a given number of times (this is denoted by the metric “stop-num”), the algorithm will treat the error FVU as being stabilized, and hence will proceed to the next step to explore the possibility of further reducing the training error by extending a new layer to the network. It should be noted that the stop-ratio and the stop-num are to be generally determined by trial and error, and one may utilize prior experience derived from simulation results to assign them.

5 Conclusions

In this paper, a new strategy for constructing a multi-hidden-layer FNN was presented. The proposed algorithm extends the constructive algorithm developed for adding hidden units to a OHL network by generating additional hidden layers. The network obtained by using our proposed algorithm has regular connections to facilitate hardware implementation. The constructed network adds new hidden units and layers incrementally only when they are needed based on monitoring the residual error that can not be reduced any further by the already existing network. We have also proposed a new type of a constructive OHL-FNN that adaptively assigns appropriate orthonormal Hermite polynomials to its generated neurons. The network generally learns as effectively as, but generalizes much better than the conventional constructive OHL networks.

References

1. Fahlman, S.E.: An Empirical Study of Learning Speed in Back-Propagation Networks. Tech. Rep., CMU-CS-88-162, Carnegie Mellon University (1988)
2. Fahlman, S.E., Lebiere, C.: The Cascade-correlation Learning Architecture. Tech. Rep., CMU-CS-90-100, Carnegie Mellon University (1991)
3. Hush, D.R., Horne, B.G.: Progress in Supervised Neural Networks. *IEEE Signal Processing Magazine*, 8–39 (January 1993)
4. Hwang, J.N., Lay, S.R., Maechler, M., Martin, R.D., Schimert, J.: Regression Modeling in Back-propagation and Projection Pursuit Learning. *IEEE Trans. on Neural Networks* 5, 342–353 (1994)
5. Korn, G.A., Korn, T.M.: *Mathematical Handbook for Scientists and Engineers*, 2nd edn. McGraw-Hill, New York (1968)
6. Kwok, T.Y., Yeung, D.Y.: Constructive Algorithms for Structure Learning in Feedforward Neural Networks for Regression Problems. *IEEE Trans. on Neural Networks* 8(3), 630–645 (1997)
7. Kwok, T.Y., Yeung, D.Y.: Objective Functions for Training New Hidden Units in Constructive Neural Networks. *IEEE Trans. on Neural Networks* 8(5), 1131–1148 (1997)
8. Lau, B., Chao, T.H.: Aided Target Recognition Processing of Mudss Sonar Data. *Proceedings of the SPIE* 3392, 234–242 (1998)
9. Linh, T.H., Osowski, S., Stodolski, M.: On-line Heart Beat Recognition Using Hermite Polynomials and Neuro-fuzzy Network. In: *IMTC/2002. Proceedings of the 19th IEEE Instrumentation and Measurement Technology Conference*, Anchorage, AK, USA, vol. 1, pp. 165–170 (2002)
10. Pilato, G., Sorbello, F., Vassallo, G.: Using the Hermite Regression Algorithm to Improve the Generalization Capability of a Neural Network. In: *Neural Nets WIRN Vietri 1999. Proceedings of the 11th Italian Workshop on Neural Nets*, Salerno, Italy, pp. 296–301 (1999)
11. Rasiyah, A.I., Togneri, R., Attikiouzel, Y.: Modeling 1-d Signals Using Hermite Basis Functions. *IEE Proc. -Vis. Image Signal Process* 144(6), 345–354 (1997)

Improved Learning Algorithms of SLFN for Approximating Periodic Function*

Fei Han

Hefei Institute of Intelligent Machines, Chinese Academy of Sciences, P.O. Box 1130,
Hefei Anhui 230031, China
hanfei1976@163.com

Abstract. In this paper, three improved Extreme Learning Machines (ELMs) are proposed to approximating periodic function. According to Fourier series expansion theory, the hidden neurons activation functions in the improved ELM are a class of sine and cosine functions. In addition, the improved ELM analytically determines the output weights of neural networks. In theory, the new algorithm tends to provide the best approximation performance at extremely fast learning speed. The proposed ELMs have better approximation accuracies and faster convergence rate than traditional ELM and gradient-based learning algorithms. Finally, experimental results are given to verify the efficiency and effectiveness of the proposed ELMs.

1 Introduction

Most traditional learning algorithms for feedforward neural networks (FNN) use backpropagation (BP) algorithm to derive the updated formulae of the weights [1]. However, these learning algorithms have the following major drawbacks that need to be improved. First, they are apt to be trapped in local minima. Second, they have not considered the network structure features as well as the involved problem properties, thus their generalization capabilities are limited [2-7]. Finally, since gradient-based learning is time-consuming, they converge very slowly [8-9].

In the literature [8-9], a learning algorithm for a single hidden layered FNN (SLFN) called as ELM was proposed to solve the problem caused by gradient-based algorithms. ELM randomly chooses the input weights and analytically determines the output weights of SLFN. ELM has much better generalization performance with much faster learning speed. However, ELM does not consider the network structure features and the involved problem properties and its generalization performance is also limited.

In the literature [10], according to Taylor series expansion, the PELM was proposed to obtain better generalization performance. The activation functions of the hidden neurons in the PELM are polynomial functions. Since PELM requires much more time to calculate the hidden neurons outputs than ELM, it converges slower than

* This work was supported by the Initial Funding of Science Research for President Prize of Chinese Academy of Sciences in the latter half of 2006.

ELM. In the literature [11], the SCELM was proposed to approximate the functions whose period was 2π . According to the parity of the approximated function, the CELM and SELM were derived from the SCELM.

In this paper, three improved ELMs for approximating periodic function are proposed. The improved ELMs select the activation functions of hidden neurons as other class of sine and cosine functions than ones in SCELM, CELM and SELM according to the Fourier series expansion theorem. The new ELMs are mainly applied to approximate the functions whose period is $2l (l \neq \pi)$. Finally, simulated results verify the efficiency and effectiveness of the new algorithms.

2 Extreme Learning Machine

In order to find an effective solution to the problem caused by BP learning algorithm, Huang [8-9] proposed ELM. Since a feedforward neural network with single nonlinear hidden layer is capable of forming an arbitrarily close approximation of any continuous nonlinear mapping, the ELM is limited to such networks.

For N arbitrary distinct samples (x_i, t_i) , where $x_i = [x_{i1}, x_{i2}, \dots, x_{in}]^T \in R^n$, $t_i = [t_{i1}, t_{i2}, \dots, t_{im}]^T \in R^m$. The SLFN with H hidden neurons and activation function $g(x)$ can approximate these N samples with zero error means that

$$\mathbf{H} \mathbf{w} \mathbf{o} = \mathbf{T} \tag{1}$$

where

$$\mathbf{H}(wh_1, wh_2, \dots, wh_H, b_1, b_2, \dots, b_H, x_1, x_2, \dots, x_N) = \begin{bmatrix} g(wh_1x_1+b_1) & \dots & g(wh_Hx_1+b_H) \\ \vdots & \dots & \vdots \\ g(wh_1x_N+b_1) & \dots & g(wh_Hx_N+b_H) \end{bmatrix}_{N \times H}, \mathbf{w} \mathbf{o} = \begin{bmatrix} wo_1^T \\ \vdots \\ wo_H^T \end{bmatrix}_{H \times m}, \mathbf{T} = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N \times m} \tag{2}$$

where $wh_i = [wh_{i1}, wh_{i2}, \dots, wh_{in}]^T$ is the weight vector connecting the i th hidden neuron and the input neurons, $wo_i = [wo_{i1}, wo_{i2}, \dots, wo_{im}]^T$ is the weight vector connecting the i th hidden neuron and the output neurons, and b_i is the threshold of the i th hidden neuron. In order to make it easier to understand ELM, In the following, a theorem is introduced:

Theorem 2.1 [8,12]: Let there exist a matrix G such that Gy is a minimum 2-norm least-squares solution of a linear system $Ax = y$. Then it is necessary and sufficient that $G = A^+$, the Moore-Penrose generalized inverse of matrix A .

In the course of learning, first, the input weights and the hidden layer biases are arbitrarily chosen and need not be adjusted at all. Second, according to Theorem 2.1, the smallest norm least-squares solution of the Eqn. (1) is obtained as follow:

$$\mathbf{w} \mathbf{o} = \mathbf{H}^+ \mathbf{T} \tag{3}$$

From the above discussion, it can be found that the ELM has the minimum training error and smallest norm of weights. The smallest norm of weights tends to have the best generalization performance. Since the solution is obtained by an analytical method and all the parameters of SLFN need not be adjusted, ELM converges much faster than when using gradient-based algorithm.

3 The Improved Extreme Learning Machines

When the function $f(x)$ meets the conditions that the Fourier series convergence theorem requires, the Fourier series of the function $f(x)$ converges to the function $\frac{1}{2}(f(x+0) + f(x-0))$. Assume that the period of the function $f(x)$ is $2l$. This can be expressed as follows:

$$\frac{1}{2}(f(x+0) + f(x-0)) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos \frac{n\pi x}{l} + b_n \sin \frac{n\pi x}{l}) \tag{4}$$

where a_n and b_n are the corresponding Fourier coefficients.

From Eqn. (4), it can be found that the function can be expressed as the weighted sum of the sine and cosine functions. Therefore, to approximate the function $f(x)$ more accurately by the SLFN, $\phi(x)$, we also make the SLFN to be expressed as the weighted sum of the sine and cosine functions. Moreover, the transfer function of the $(2k-1)$ th hidden neuron is selected as the function $\sin \frac{k\pi x}{l}$ and the one of the $(2k)$ th hidden neuron is selected as the function $\cos \frac{k\pi x}{l}$. However, the transfer function of the $(2n+1)$ th hidden neuron is selected as $\cos(0x)$. Then, the SLFN $\phi(x)$ can be expressed as follows:

$$\phi(x) = \sum_{k=1}^n (w_{O_{2k}} \cos(\frac{k\pi(wh_{2k}x)}{l}) + w_{O_{2k-1}} \sin(\frac{k\pi(wh_{2k-1}x)}{l}) + w_{O_{2n+1}} \cos(0(wh_{2n+1}x))) \tag{5}$$

where w_{O_i} denotes the synaptic weight from the output neuron to the i th neuron at the hidden layer, and wh_i denotes the synaptic weight from the i th neuron at the hidden layer to the input neuron. The output layer is linear. The weights from the output neuron to the hidden neurons are analytically determined by Eqn. (3). This algorithm is referred to as SCELM-II. From SCELM-II, the following conclusions can be easily deduced:

Conclusion 1: Assume that the FNN, $\phi(x)$, which is expressed as Eqn. (5), is used to approximate the periodic function $f(x)$ by SLFN with SCELM-II. The period of the function is $2l$. The function $f(x)$ meets the conditions that the Fourier series convergence theorem requires, and x is in the definitional domain of the function $f(x)$. The following equation can be obtained:

$$w_{O_{2k-1}} \approx b_k, w_{O_{2k}} \approx a_k \quad k=1,2,\dots,n, w_{O_{2n+1}} \approx \frac{a_0}{2} \tag{6}$$

where a_k , b_k ($k=1,2,\dots,n$) and a_0 are Fourier coefficients of the function $f(x)$.

Proof: Comparing Eqns. (4) with Eqn. (5), we notice that $f(x) \approx \phi(x)$ and $wh_k=1$, ($k=1,2,\dots,2n$) from SCELM. So Eqn. (6) can be easily deduced. \square

Apparently, SCELM-II has two special cases. On one hand, when the periodic function $f(x)$ is even, the transfer function of the k th hidden neuron is selected as the function $\cos \frac{k\pi}{T}$ and the one of the $(n+1)$ th hidden neuron is selected as one. This new algorithm is called the Cosine ELM-II (CELM-II). On the other hand, when the periodic function $f(x)$ is odd, the transfer function of the k th hidden neuron is selected as the function $\sin \frac{k\pi}{T}$ and the one of the $(n+1)$ th hidden neuron is selected as one. This new algorithm is called the Sine ELM-II (SELM-II).

In the above improved ELMs, because the output weights are analytically determined, the learning speed of the improved ELMs is thousands of times faster than that of a gradient-based algorithm. Similarly, according to Eqn. (3), since the smallest norm least-squares solution is obtained, the improved ELM tends to have the better generalization performance. Finally, compared with ELM, in that these improved ELMs incorporate architectural constraints from *a priori* information into SLFN, they have better approximation performance for approximating a function.

4 Experimental Results

First, to demonstrate the improved approximation performance and fast convergence rate of the improved ELMs with respect to the BP algorithm, ELM, PELM, SCELM, CELM and SELM, in the following we shall conduct the experiment with a periodic function. The periodic function is defined as $y = (1 - (40x/\pi) + 2(40x/\pi)^2 - 0.4(40x/\pi)^3)e^{-x/2}$ at one period $[-2, 2]$. The activation functions of the neurons in all layers for BP algorithm are tangent sigmoid functions and the ones of the hidden neurons for ELM are sigmoid functions. As for BP algorithm, the number of the hidden neurons is 12, whereas the one of the improved ELMs and ELM is 21.

Assume that the number of the total training data is 81, which are selected from $[0, 2]$ at identically spaced intervals. Similarly, 80 testing samples are selected from $[0.0125, 1.9875]$ at identically spaced intervals. As a result, the approximation error of the testing samples with the SCELM-II, CELM-II and SELM-II is shown in Fig. 1.

In order to statistically compare the approximation accuracies and CPU time for the functions with the above learning algorithms, we conducted the experiments thirty times for each algorithm, and the corresponding results are summarized in Table 1.

From the above results, it can be drawn the conclusions as follows:

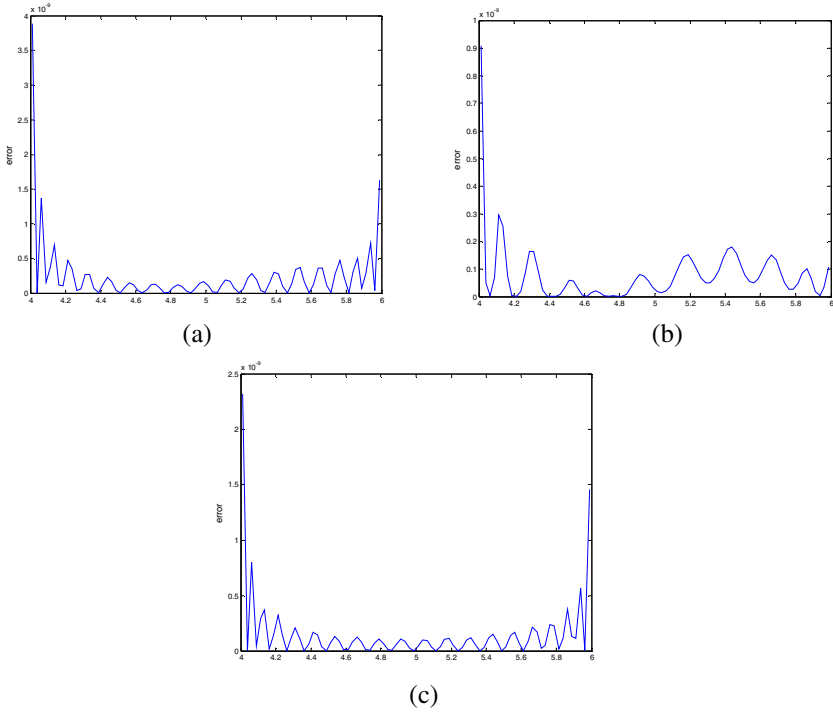


Fig. 1. The approximation errors of the testing samples for approximating the function $y = (1 - (40x/\pi) + 2(40x/\pi)^2 - 0.4(40x/\pi)^3)e^{-x/2}$ using three improved ELMs. (a) SCELM-II; (b) CELM-II; (c) SELM-II

Table 1. The approximation accuracies and CPU time for approximating the function $y = (1 - (40x/\pi) + 2(40x/\pi)^2 - 0.4(40x/\pi)^3)e^{-x/2}$ with nine algorithms

LA	Training error	Testing error	CPU time
BP	5.0428e-4	3.8669e-4	34.7810s
ELM	8.9360e-6	8.7460e-6	0.0480s
PELM	6.2550e-6	5.9200e-6	0.2563s
SCELM	4.4792e-10	4.6714e-10	0.0450s
CELM	6.4065e-10	6.7855e-10	0.0432s
SELM	2.3156e-10	2.4351e-10	0.0443s
SCELM-II	4.5522e-10	5.1284e-10	0.0460s
CELM-II	3.4787e-10	3.7045e-10	0.0435s
SELM-II	2.4885e-10	2.1929e-10	0.0433s

First, the approximation accuracies of all ELMs are much better than that of the BP algorithm, because the testing errors of all ELMs are much less than that of BP algorithm. This result rests in the fact that all ELMs obtain the smallest norm least-squares solution through Eqn. (3).

Second, all ELMs converge much faster than BP algorithm. This is because all ELMs obtain the solution by analytical means, whereas BP algorithm obtains the solution through thousands of iterative calculations.

Third, in all ELMs, the approximation accuracies of the ELM and PELM are worse than the ones of other ELMs. Moreover, the ELM and PELM converge slower than other ELMs. Since the PELM spends much more time to calculate the hidden neurons outputs than other ELMs, it converges slowest in all ELMs.

Finally, there is not much difference between the improved ELMs (SCELM, CELM and SELM) in literature [11] and the other improved ones (SCELM-II, CELM-II, SELM-II) in this paper in approximation accuracy and convergence rate. However, SCELM, CELM and SELM are applied to approximate the functions whose period is 2π , whereas the SCELM-II, CELM-II, SELM-II are used to approximate the functions whose period is $2l$.

Figure 2 shows the relation between the testing error and the values of the parameter l for improved ELMs proposed in this paper for approximating the above function. It can be seen that the ideal values of the parameter l for the SCELM-II, CELM-II, SELM-II is at $2 \sim 5$.

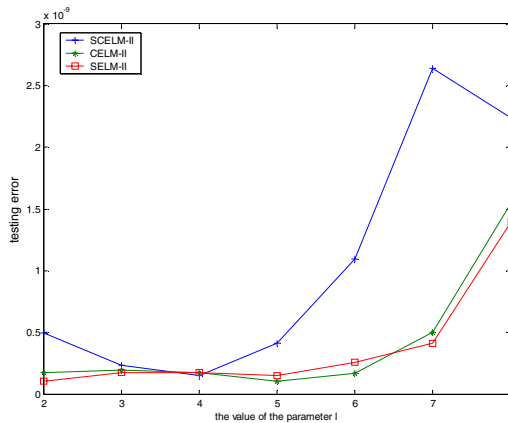


Fig. 2. The relation between the testing error and the value of the parameter l with three improved ELMs for approximating the function $y = (1 - (40x/\pi) + 2(40x/\pi)^2 - 0.4(40x/\pi)^3)e^{-x/2}$

5 Conclusions

In this paper, three improved ELMs were proposed for function approximation problem. The improved ELMs are applied to approximate the functions whose period is $2l$. The architectural constraints are extracted from *a priori* information of the approximated function based on Fourier series expansion. The architectural constraints are realized by selecting the activation functions of the hidden neurons in the improved ELMs as a class of sine and cosine functions. Furthermore, the new algorithms

analytically determine the output weights of SLFN through simple generalized inverse operation of the hidden layer output matrices according to ELM. Therefore, the new ELMs has much better approximation accuracies and faster convergent rate than the traditional gradient-based learning algorithms and ELM. Finally, simulated results were given to verify the efficiency and effectiveness of the improved ELMs.

References

1. Ng, S.C., Cheung, C.C., Leung, S.H.: Magnified Gradient Function with Deterministic Weight Modification in Adaptive Learning. *IEEE Transactions on Neural Networks* 15(6), 1411–1423 (2004)
2. Baum, E., Haussler, D.: What Size Net Gives Valid Generalization? *Neural Comput.* 1(1), 151–160 (1989)
3. Huang, D.S.: A Constructive Approach for Finding Arbitrary Roots of Polynomials by Neural Networks. *IEEE Transactions on Neural Networks* 15(2), 477–491 (2004)
4. Huang, D.S., Chi, Zheru: Finding Roots of Arbitrary High Order Polynomials Based on Neural Network Recursive Partitioning Method. *Science in China Ser. F Information Sciences* 47(2), 232–245 (2004)
5. Huang, D.S., Ip, H.H.S., Chi, Z.: A Neural Root Finder of Polynomials Based on Root Moments. *Neural Computation* 16(8), 1721–1762 (2004)
6. Huang, D.S., Ip, H.H.S., Chi, Zheru, Wong, H.S.: Dilation Method for Finding Close Roots of Polynomials Based on Constrained Learning Neural Networks. *Physics Letters A* 309(5-6), 443–451 (2003)
7. Karras, D.A.: An Efficient Constrained Training Algorithm for Feedforward Networks. *IEEE Trans. Neural Networks* 6(6), 1420–1434 (1995)
8. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme Learning Machine: A New Learning Scheme of FNN. In: 2004 International Joint Conference on Neural Networks (IJCNN 2004), Budapest, Hungary, July 25-29, pp. 985–990 (2004)
9. Huang, G.B., Siew, C.K.: Extreme Learning Machine with Randomly Assigned RBF Kernels. *International Journal of Information Technology* 11(1), 16–24 (2005)
10. Han, F., Lok, Tat-Ming, Lyu, Michael, R.: A New Learning Algorithm for Function Approximation Incorporating a priori Information into Extreme Learning Machine. In: Wang, J., Yi, Z., Žurada, J.M., Lu, B.-L., Yin, H. (eds.) *ISNN 2006. LNCS*, vol. 3971, pp. 631–636. Springer, Heidelberg (2006)
11. Han, F., Huang, D.S.: Improved Extreme Learning Machine for Function Approximation by Encoding a priori Information. *Neurocomputing* 69(16-18), 2369–2373 (2006)
12. Serre, D.: *Matrices: Theory and Application*, p. 147. Springer, New York (2002)

Some Progress of Supervised Learning

Chunyang Su¹, Shifei Ding^{1,2}, Weikuan Jia³, Xin Wang¹, and Xinzheng Xu¹

¹ School of Computer Science and Technology, China University of Mining and Technology,
Xuzhou 221008 China

² Key Laboratory of Intelligent Information Processing, Institute of Computing Technology,
Chinese Academy of Sciences, Beijing 100080 China

³ College of Plant Protection, Shandong Agricultural University, Taian 271018, China
dingsf@cumt.edu.cn, dingshifei@sina.com

Abstract. Supervised learning is very important in machine learning. In this paper we discuss some progress of supervised learning. At first, we introduce the basic concept and methods of supervised learning; then explain several typical algorithms of supervised learning in details, the algorithms covered are Bayesian networks, decision tree, k-nearest neighbor, supervised manifold learning and support vector machines; at last we point out several developing directions of supervised learning.

1 Introduction

As a broad subfield of artificial intelligence, machine learning is concerned with the design and development of algorithms and techniques that allow computers to "learn". The major focus of machine learning research is to extract information from data automatically, by computational and statistical methods. Every instance in any dataset used by machine learning algorithms is represented using the same set of features. The features may be continuous, categorical or binary. If instances are given with known labels then the learning is called supervised, in contrast to unsupervised learning [1]. Semi-supervised learning combines both labeled and unlabeled examples to generate an appropriate function or classifier. Another kind of machine learning is reinforcement learning [2], in which the algorithm learns a policy of how to act given an observation of the world. Every action has some impact in the environment, and the environment provides feedback that guides the learning algorithm.

Supervised learning is a machine learning technique for creating a function from training data. The training data consist of pairs of input objects, and desired outputs. The output of the function can predict a class label of the input object. The task of the supervised learner is to predict the value of the function for any valid input object after having seen a number of training examples. To achieve this, the learner has to generalize from the presented data to unseen situations in a "reasonable" way. Classifier performance depends greatly on the characteristics of the data to be classified. There is no single classifier that works best on all given problems. Determining a suitable classifier for a given problem is still more an art than a science.

2 Supervised Learning Algorithms

2.1 Bayesian Networks (BNs)

Consider a problem defined by a set of variables $X = \{X_1, \dots, X_n\}$. The Bayesian network structure S is a directed acyclic graph (DAG) and the nodes in S are in one-to-one correspondence with the features X . The presence of an arc between two variables expresses the existence of dependence between them, which is quantified by the conditional distribution assigned to the nodes. The two major tasks in learning a BN are: learning the graphical structure, and then learning the parameters for that structure. There are two ways to view a BN, each suggesting a particular approach to learning. First, a BN is a structure that encodes the joint distribution of the attributes. This suggests that the best BN is the one that best fits the data, and leads to the scoring-based learning algorithms, that seek a structure that maximizes the Bayesian, MDL or Kullback-Leibler (KL) entropy scoring function [3]. Second, the BN structure encodes a group of conditional independence relationships among the nodes, according to the concept of d-separation. This suggests learning the BN structure by identifying the conditional independence relationships among the nodes.

Bayesian Networks will reduce to simple multiplication in our classification context, when all the values of the dataset attributes are known. But BNs have an inherent limitation. Bayesian network inference is NP-hard in general when exploring a previously unknown network. Given a problem described by n features, the number of possible structure hypotheses is more than exponential in n . If the structure is unknown, one approach is to introduce a scoring function that evaluates the “fitness” of networks with respect to the training data, and then to search for the best network according to this score. Several researchers have shown experimentally that the selection of a single good hypothesis using greedy search often yields accurate predictions. BNs take into account prior information about a given problem, in terms of structural relationships among its features. BN classifiers are not suitable for datasets with many features.

2.2 Decision Tree Learning

When encountering the following conditions: a. Instances describable by attribute–value pairs; b. Target function is discrete valued; c. Disjunctive hypothesis may be required; d. possibly noisy training data, we should consider decision trees. This is because decision trees have several advantages: It is simple to understand and interpret. It uses a white box model. It is possible to validate a model using statistical tests. That makes it possible to account for the reliability of the model. It is robust, and performs well with large data in a short time. It is simpler and robust to incomplete and noise data due to post-pruning techniques [4].

The feature that best divides the training data would be the root node of the tree. There are numerous methods for finding the feature that best divides the training data such as information gain and gini index. However, a majority of studies have concluded that there is no single best method [5]. Overfitting can occur with noisy training examples, and also when small numbers of examples are associated with leaf nodes.

There are two common approaches that decision tree induction algorithms can use to avoid overfitting training data: i) Stop growing when data split not statistically significant, ii) Prune the induced decision tree.

Many researchers have found that decision tree learning such as ID3, C4.5 and CART perform well in data classification. The most well-know algorithm for building decision trees is the C4.5 [6]. C4.5 is an extension of Quinlan's earlier ID3 algorithm [7]. Ruggieri presented an analytic evaluation of the runtime behavior of the C4.5 algorithm, which highlighted some efficiency improvements. Based on this analytic evaluation, he implemented a more efficient version of the algorithm, called EC4.5 [8]. He argued that his implementation computed the same decision trees as C4.5 with a performance gain of up to five times. Olcay and Onur [9] show how to parallelize C4.5 algorithm in three ways: (i) feature based, (ii) node based (iii) data based manner. Baik and Bala [10] presented preliminary work on an agent-based approach for the distributed learning of decision trees. Decision trees tend to perform better when dealing with discrete/categorical features.

2.3 K-Nearest Neighbor (KNN) Algorithm

KNN is part of supervised learning that has been used in many applications in the field of data mining, statistical pattern recognition, image processing, etc. KNN is robust to noisy training data and effective if the training data is large. However, it needs to determine value of parameter K, it is not clear which type of distance to use and which attribute to use to produce the best results, and the computation cost is quite high because we need to compute distance of each query instance to all training samples. Some indexing (e.g. K-D tree) may reduce this computational cost.

KNN is a supervised learning algorithm where the result of new instance query is classified based on majority of K-nearest neighbor category. The purpose of this algorithm is to classify a new object based on attributes and training samples. The classifiers do not use any model to fit and only based on memory. Given a query point, we find K number of objects or training points closest to the query point. The classification is using majority vote among the classification of the K objects. Any ties can be broken at random. KNN algorithm used neighborhood classification as the prediction value of the new query instance. The quality of KNN's generalization therefore depends on which instances are deemed least distant, which is determined by its distance function. Wettschereck et al. [11] showed that the performance of KNN was not sensitive to the exact choice of k when k was large. Breiman [12] reported that the stability of nearest neighbor classifiers distinguishes them from decision trees and some kinds of neural networks.

2.4 Supervised Manifold Learning

Manifold learning is a kind of unsupervised learning [13]. It has the advantage of good dimension reduction ability. First, manifold learning stresses that the high-dimensional data are generated by the inner low-dimensional variables. Second, manifold learning supposes that the dataset contained by each category form a manifold, the dataset of

different categories lie in different manifolds. This is different from the traditional assumptions (data lie in linear space), and makes the supervised learning based on manifold feasible. Supervised learning and manifold learning may be opposite. Most of the supervised manifold learning algorithms are the compromise of the two. In recent years some supervised manifold learning algorithms have been proposed. Roweis et al. proposed Locally Linear Embedding (LLE) algorithm which is a new dimension reduction technique aiming at nonlinear data [14]. It can make the data after dimension reduction keep original topology. Ridder et al. proposed a supervised manifold learning algorithm and its improvements [15]. The traditional LLE algorithm searches the k nearest neighbor points according to the Euclidean distance between sample points while the Supervised Locally Linear Embedding (SLLE) adding the category information of the sample points. They pointed out that when the data were low-dimensional, the SLLE algorithm that used supervised reduction technique didn't improve obviously on the precision, but it would be different when the dataset were high-dimensional. SLLE algorithm nonlinearly maps the data to the sub-spaces with strong discriminant ability, so the simple classifiers can get good recognition performance. SLLE can be seen as the popularization of the KNN algorithm. Zhang et al. [16] designed the ULLELDA algorithm based on Gabor and integration in the case of single sub-space, and proposed MUSNACAL algorithm and the corresponding integrated algorithm when there're several manifold structures. By a series of comparative experiments, they tested and verified the effectivity of these classifiers and the dimensionality reduction methods.

2.5 Support Vector Machines (SVMs)

The number of support vectors selected by the SVM learning algorithm is usually small. For this reason, SVMs are well suited to deal with learning tasks where the number of features is large. Most real-world problems involve nonseparable data for which no hyperplane exists that successfully separates the positive from negative instances in the training set. One solution to the inseparability problem is to map the data onto a higher dimensional space and define a separating hyperplane there. This higher-dimensional space is called the transformed feature space. With an appropriately chosen transformed feature space of sufficient dimensionality, any consistent training set can be made separable.

Training the SVM is done by solving N -dimensional quadratic programming (QP) problem, where N is the number of samples of the training dataset. Solving this problem in standard QP methods involves large matrix operations, as well as time-consuming numerical computations. We usually use some decomposition methods to decompose the problems into a series of small-scale quadratic optimal problems. The commonly used methods are Chunking Algorithms [17], Decomposition Algorithms [18] and Sequential Minimal Optimization (SMO). These algorithms are all based on the fact that it won't affect the solution of the original problems when excluding the training samples of non-support vectors. The goal of Chunking Algorithms is to exclude the non-support vectors step by step, so as to decompose the original quadratic optimal problems to several small quadratic optimal problems. This will ease

the need for the memory. SMO is one special case of Decomposition Algorithms; it is the extreme case that decreasing the sample sets to two samples. SMO is a simple algorithm that can, relatively quickly, solve the SVM QP problem without any extra matrix storage and without using numerical QP optimization steps at all [19]. SMO decomposes the overall QP problem into QP sub-problems. Keerthi and Gilbert [20] suggested two modified versions of SMO that are significantly faster than the original SMO in most situations. Currently, SMO is thought as a fast algorithm for solving optimization problems.

3 Prospect

In this paper, we roughly discuss several best-known supervised learning algorithms.

When dealing with machine learning classification, the most important problem is not whether a learning algorithm is superior to others, but under what conditions a particular algorithm can perform better than others. After better understanding the strengths and limitations of each algorithm, we can consider integrating several algorithms together to solve a problem which will utilize the strengths of one method to complement the weaknesses of another. The goal of integrated algorithms is to generate more certain, precise and accurate system results. However, the integrated methods may increase the storage space and the computation time. In many practical problems, the datasets are large, so distributed machine learning is also a research direction of supervised learning. It divides the dataset into several subsets, learns from these subsets and combines the results. The research of the methods for the construction of good ensembles of classifiers is also an active research area in supervised learning. Mechanisms that are used to build ensemble of classifiers include: 1) using different subsets of training data with a single learning method; 2) using different training parameters with a single training method; 3) using different learning methods. However, besides the above research directions, supervised learning still has many other research directions. For example, the overfitting control of the learning methods, the compromise between the intelligibility of the algorithms and the classification precision, and the methods for expanding the applied area of supervised learning, etc.

Acknowledgements. This work is supported by the National Natural Science Foundation of China under Grant no.40574001 and no.60675010, the 863 National High-Tech Program under Grant no. 2006AA01Z128, and the Opening Foundation of Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences under Grant no.IIP2006-2.

References

1. Jain, A.K.: Data Clustering: A Review. *ACM Computing Surveys* 31, 264–323 (1999)
2. Leslie, P.K.: Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research* 4, 237–285 (1996)

3. Cooper, G.F., Herskovits, E.: A Bayesian Method for the Induction of Probabilistic Networks from Data. *Machine Learning* 9, 309–347 (1992)
4. Pal, M., Mather, P.M.: Decision Tree Based Classification of Remotely Sensed Data. In: Proceedings of 22nd Asian Conference on Remote Sensing, Singapore, pp. 245–248 (2001)
5. Murthy: Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey. *Data Mining and Knowledge Discovery* 2, 345–389 (1998)
6. Quinlan, J.R.: *Discovering Rules by Induction from Large Collections of Examples*. Expert Systems in the Microelectronic age, pp. 168–201 (1979)
7. Quinlan, J.R.: *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Francisco (1993)
8. Ruggieri, S.: Efficient C4.5. *IEEE Transactions on Knowledge and Data Engineering* 14, 438–444 (2001)
9. Olcay, T.Y., Onur, D.: Parallel Univariate Decision Trees. *Pattern Recognition Letters* 28, 825–832 (2007)
10. Baik, S., Bala, J.: A Decision Tree Algorithm for Distributed Data Mining: Towards Network Intrusion Detection. In: Laganá, A., Gavrilova, M.L., Kumar, V., Mun, Y., Tan, C.J.K., Gervasi, O. (eds.) *ICCSA 2004*. LNCS, vol. 3046, pp. 206–212. Springer, Heidelberg (2004)
11. Wettschereck, D.: A Review and Empirical Evaluation of Feature Weighting Methods for a Class of Lazy Learning Algorithms. *Artificial Intelligence Review* 10, 1–37 (1997)
12. Breiman, L.: Bagging Predictors. *Machine Learning* 24, 123–140 (1996)
13. Saul, L.K., Roweis, S.T.: Think Globally, Fit Locally: Unsupervised Learning of Low Dimensional Manifolds. *Journal of Machine Learning Research* 4, 119–125 (2003)
14. Sam, T.R., Lawrence, K.S.: Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 2323–2326 (2000)
15. Dick, d.R., Olga, K., Oleg, O.: Supervised Locally Linear Embedding. In: Kaynak, O., Alpaydm, E., Oja, E., Xu, L. (eds.) *ICANN 2003 and ICONIP 2003*. LNCS, vol. 2714, pp. 333–341. Springer, Heidelberg (2003)
16. Zhou, Z.H., Wang, J.: *Machine Learning and Its Applications 2007*. Tsinghua University Press, Beijing (2007)
17. Vapnik, V.: *Estimation of Dependences Based on Empirical Data*. Springer, Berlin (1982)
18. Osuna, E.E.: *Support Vector Machines: Training and Applications*. Technical Report, AIM 1602, Cambridge, MA: MIT Artificial Intelligence Laboratory (1996)
19. Platt, J.: Using Sparseness and Analytic QP to Speed Training of Support Vector Machines. In: *Advances in Neural Information Processing Systems*. MIT Press, Cambridge (1999)
20. Keerthi, S., Gilbert, E.: Convergence of a Generalized SMO Algorithm for SVM Classifier Design. *Machine Learning* 46, 351–360 (2000)

Supervised Gravitational Clustering with Bipolar Fuzzification

Umut Orhan¹, Mahmut Hekim², and Turgay Ibrikli³

^{1,2} Electronics and Computer Department, Gaziosmanpasa University 60250, Tokat, Turkiye

³ Electrical and Electronics Engineering, Cukurova University 01330, Adana, Turkiye
umutorhan@mail.gop.edu.tr, mhekim@gop.edu.tr,
ibrikli@cukurova.edu.tr

Abstract. Data clustering is an important part of cluster analysis. Numerous semi-supervised or supervised clustering algorithms based on various theories have been developed, and new clustering algorithms continue to appear in the literature. The problem of common supervised clustering is to train a clustering algorithm to produce desirable clusters and complete clusters over datasets and learn how to cluster future sets of objects. In this paper, we have proposed an algorithm called Supervised Gravitational Clustering based on bipolar fuzzification. Traditional supervised clustering methods identify *class-uniform clusters*; but the offered method identifies *class-multiform clusters* with high probability densities. For this aim we have proposed two approaches: common effect and maximal effect. The first, *common effect approach*, calculates total effect of all class-centers over searching point. Also, this approach is basis for mapping of novel method. The second, *maximal effect approach*, determines class-centers with the strongest effect over searching point.

Keywords: supervised gravitational clustering; bipolar fuzzy; K-means; fuzzy C-means.

1 Introduction

The expectation of a clustering algorithm is that the objects in the same cluster should be more similar than the objects in different cluster. A clustering algorithm may not produce desirable clusters without additional information from the user. This information is manual adjustment of the algorithm or similarity measure. The clustering is an unsupervised learning technique using an error function; whereas the classification is a supervised learning technique using a probability function [1], [2], [3], [4], [5], [6], [7], [8].

There has been some work that has some similarity with our research under the heading of supervised clustering. The idea of supervised clustering is to improve a clustering algorithm by using class information during the clustering process. The supervised clustering process optimizes class purity in addition to the traditional objectives of a clustering algorithm. Class purity means that classified objects belonging to the same class should be assigned to the same cluster. The existing research on

supervised clustering can be subdivided into two major groups: similarity based methods and search-based methods [9], [10], [11], [12]. The modified distance function combined with knowledge regarding classified objects is used in similarity-based method. Search-based methods, on the other hand, modify the clustering algorithm but do not change the distance function. In literature, there has been some research which obtain some constraints by training classified objects [10], [11]. Klein proposes a shortest path algorithm by modifying the Euclidian distance function based on prior knowledge in K-means clustering algorithm [12]. Demiriz proposes an evolutionary clustering algorithm to obtain clusters, in which the search process minimizes cluster dispersion and impurity [13]. Sinkkonen proposes an approach called discriminative clustering that minimizes distortion within clusters [14]. Distortion, in their context, represents information loss. This technique produces clusters belonging to a single class that are internally as possible as homogeneous with respect to conditional distributions. Similarly, Tishby introduced the information bottleneck method [15]. Based on that method, an aggregate clustering algorithm is proposed for minimizing information loss with respect to conditional distribution. In this paper, we study a method based on gravitational computation which detects the clusters in each class by using bipolar fuzzy.

The paper is organized as follows. In Section 2, the structure of the new supervised clustering is presented. Section 3 shows results for some artificial and real datasets. Our results are discussed in Section 4. Finally, the conclusions are given in Section 5.

2 Supervised Gravitational Clustering with Bipolar Fuzzification

Common supervised clustering is based on class-uniform clusters; but the method called Supervised Gravitational Clustering with Bipolar Fuzzification (SGF) identifies class-multiform clusters which are inspired by contour line mapping. The contour line mapping makes line on a topographical map which indicates points of the same altitude. Suppose that one of the classes is lands; another is seas. The clusters belonging to the *land-class* will be called as *hill-clusters*; the clusters belonging to the *sea-class* will be called as *trench-clusters*. By using density based fuzzy memberships, the densest point of each cluster is selected as cluster center (CC).

The densest point in each hill-cluster is selected as *peak-CC*, and the densest point in each trench-cluster is selected as *depth-CC*.

Each peak-CC affects its surroundings by proportion of its altitude and it presents its surroundings as a hill-cluster. Similarly, each depth-CC affects a point by proportion of its altitude and it presents its surroundings as a trench-cluster. The points far from a CC are less affected by that CC; on the contrary, the points closer to a CC are more affected by that CC. In order to see whether searching point is below or above sea level, the effect of each known hill-cluster and trench-cluster must be determined. For this aim, densities of all peak-CCs and depth-CCs are calculated by using distances between searching point and each CC. The effects are directly proportion to altitude of all clusters, and inversely proportion to their distances.

Each point in the space affects each other, but massive points affect weak points more than the others. Therefore, the mass should be taken account of finding densities.

According to the law of gravitation, the attraction between two objects is directly proportional to the product of their masses and inversely proportional to the square of the distance between them. By inspired of the law, we suppose that each CC is a mass gravity center and each data point's mass is constant and thus the law equation m/d^2 is transformed to $1/d^2$ and the density of j th data point is calculated by using Equation 1.

$$m_j = \sum_{i=1}^n \frac{1}{\|x_j - x_i\|^2} \quad (1)$$

where n is the sample size of dataset and $\|x_j - x_i\|$ is the distance between data points x_i and x_j . At the first iteration of the algorithm, the gravitational densities of each point to each others are calculated by using Equation 1 [7]. The maximum density of each class is selected as CCs. One of them is peak-CC, and the other is depth-CC. Then the densities of the misclassified data points D_i are computed by using either Equation 2 for common effect approach or Equation 3 for maximal effect approach. *Common effect approach* calculates CCs by using Equation 2 which maps dataset by the summation of all CC effects over searching point. *Maximal effect approach* computes CCs which has the strongest effect by using Equation 3.

$$D_{i \text{ common}} = \sum_{j=1}^C \frac{m_j}{\|CC_j - x_i\|^2} \quad (2)$$

$$D_{i \text{ maximal}} = \text{Max}_{j=1}^C \frac{m_j}{\|CC_j - x_i\|^2} \quad (3)$$

where C is the number of detected clusters and CC_j is j th cluster center.

Suppose that the effect of each hill-cluster is positive and the effect of each trench-cluster is negative, then the total of effects determines whether the searching point is below or above sea level. For determining the altitude of searching point, that point is defined by membership of each CC. The bipolar fuzzy membership used for this aim is defined below.

Each point belongs to a cluster with a membership degree in classical fuzzy. In this paper, suppose that fuzzy membership is bipolar which the memberships are positive or negative signed. When the sum of memberships is positive signed, searching point belongs to land-class. Otherwise, it belongs to sea-class. Namely, positive value means above sea level, whereas negative value means below sea level.

Equation 4 computes μ_{ij} the membership values of j th data point to i th cluster. The sum of unsigned membership values equals to 1, as in classical fuzzy. The sum of signed membership values computed by Equation 4 describes the altitude of searching point.

$$\mu_{ij} = \frac{\tilde{\lambda}}{\|c_i - x_j\|^2}, \frac{1}{\tilde{\lambda}} = \sum_{i=1}^k \frac{1}{\|c_i - x_j\|^2} \tag{4}$$

where $\tilde{\lambda}$ is normalization coefficient and k is the number of data points belonging to investigated cluster. Our algorithm in this paper consists of steps below.

- Initial:* Gravitational density of each point is calculated by Equation 1. The maximum densities are selected as CCs. The point with maximum density in *land-class* is peak-CC, and the point with maximum density in *sea-class* is depth-CC.
- Step 1.* Detect data points belonging to each CC and update the densities of misclassified data points by Equation 2 (for common effect) or Equation 3 (for maximal effect).
- Step 2.* Stop the algorithm if the success of result is sufficient.
- Step 3.* Select the densest point as CC among misclassified data points.
- Step 4.* Stop the algorithm if a new CC can not be chosen.
- Step 5.* Compute the membership values using Equation 4, and select the maximum absolute value of densities as CC. Go to Step 1.

3 Numerical Examples

In this section it has been presented four artificial and five real datasets examples to provide several viewpoints into this new approach. For simplicity, we have prepared two dimensional matrices. The figures below show that comparison of mentioned approaches above and the cluster mapping of datasets.

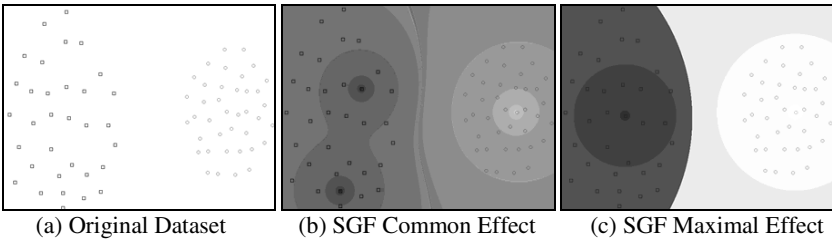


Fig. 1. Comparison of two approaches of SGF method for a basic dataset

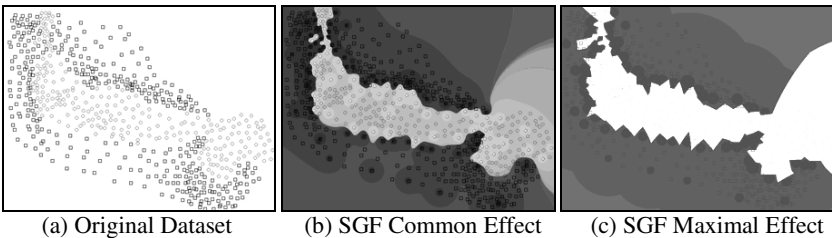


Fig. 2. Comparison of two approaches of SGF method for a nonlinear dataset

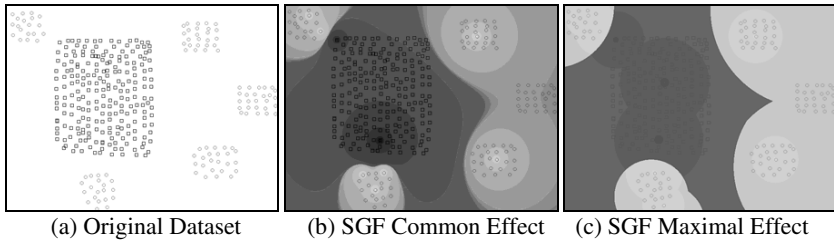


Fig. 3. Comparison of two approaches of SGF method for a multi-class dataset

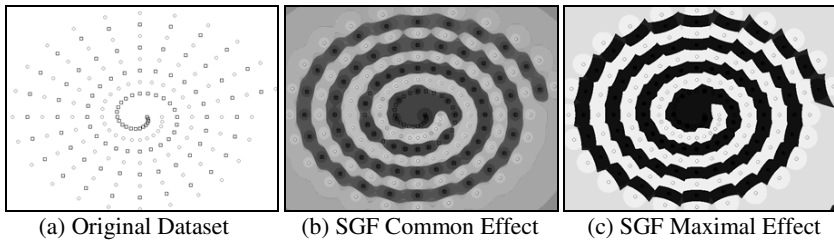


Fig. 4. Comparison of two approaches of SGF method for a spiral dataset

We applied offered algorithms to a benchmark consisting of the datasets listed in Table 1. All datasets were obtained from University of California at Irving Machine Learning Repository [16]. Because of evaluating only binary classification problem in this paper, we have chosen datasets with two classes having frequently been used in literature. Table 2 lists three widely used measures defined by previous related works.

Table 1. Summary of datasets

Dataset Name	Instances	Attributes
Haberman's Survival	306	3
Ionosphere	351	34
Pima Indians Diabetes	768	8
SPECT Heart	267	44
Wisconsin Breast Cancer	683	10

Table 2. The definition of measures employed in the study

Measure	Abbreviation	Equation
Sensitivity	Sens	$\frac{TP}{TP + FN}$
Specificity	Spec	$\frac{TN}{TN + FP}$
Total Classification Accuracy	Acc	$\frac{TP + TN}{TP + TN + FP + FN}$

The definition of the abbreviations used: *TP* is the number of correctly classified disordered residues; *FP* is the number of ordered residues incorrectly classified as disordered; *TN* is the number of correctly classified ordered residues; and *FN* is the number of disordered residues incorrectly classified as ordered.

Sensitivity is the number of true positive decisions divided by the number of actually positive cases. Specificity is the number of true negative decisions divided by the number of actually negative cases. Total classification accuracy is the number of correct decisions divided by the total number of cases [16]. Figures below shows the relation between the success and the iteration for datasets listed in Table 1.

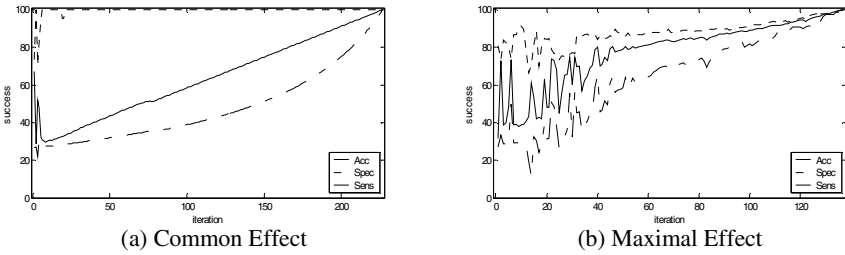


Fig. 5. Relation between the success and the iteration for Haberman's Survival dataset

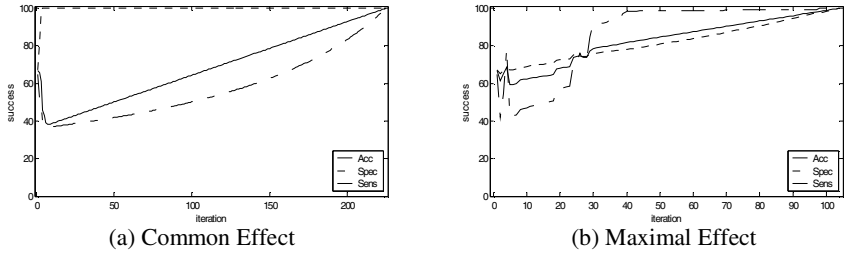


Fig. 6. Relation between the success and the iteration for Ionosphere dataset

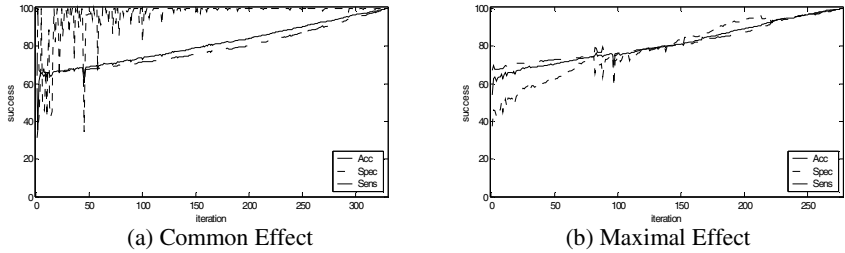


Fig. 7. Relation between the success and the iteration for Pima Indians Diabetes dataset

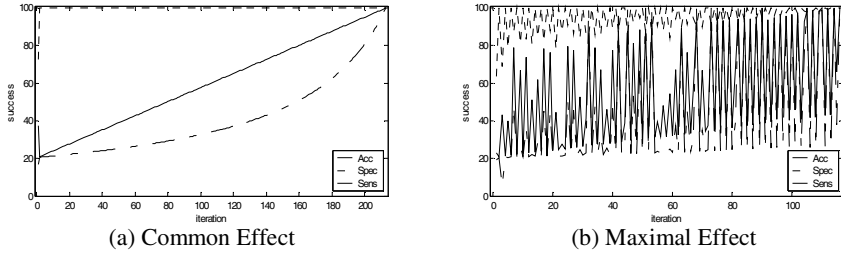


Fig. 8. Relation between the success and the iteration for SPECT Heart dataset

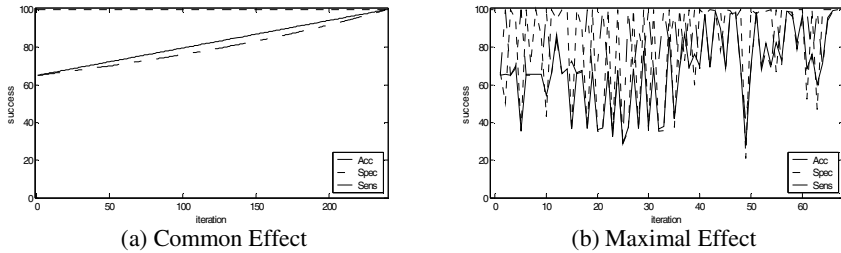


Fig. 9. Relation between the success and the iteration for Wisconsin Breast Cancer dataset

4 Results and Discussion

All figures show the relation between the success and the iteration for datasets by using the validation measurements *Acc*, *Spec* and *Sens*. The overfitting in common effect approach in Figure 5, 6, 8 and 9 starts virtually in the beginning of iterations; but the overfitting in maximal effect approach in Figure 5, 6 and 7 starts in the middle of figure. The oscillations seen in figures are regions where non-noisy clusters are detected by method.

Figure 8 and 9 shows that Wisconsin Breast Cancer and SPECT Heart datasets include intermixed classes. So, the method can hardly learn without overfitting for this kind of homogeneous datasets. Figure 5, 6 and 7 shows that Haberman's Survival, Ionosphere and Pima Indians Diabetes datasets are suitable for clustering. The method can detect easily starting of overfitting for this kind of heterogeneous datasets.

5 Conclusion

A novel method called *Supervised Gravitational Fuzzy Clustering with Bipolar Fuzzification* has been studied in this paper. The aim of traditional supervised clustering is to identify class-uniform clusters; but this paper focuses on class-multiform clusters, which proposes two approaches: common effect and maximal effect.

Common supervised clustering is based on class-uniform clusters; but SGF method identifies class-multiform clusters which are inspired by contour line mapping. In this method, each peak-CC and depth-CC affect their surroundings by proportion of their altitudes. We propose two approaches based on the law of gravity in order to see whether

a point is below or above sea level. Then, we offer bipolar fuzzy membership function. The sum of signed membership values describes altitude or deepness of points.

In future work, the overfitting region can be estimated by using derivative of figures. If the derivative variation of relation between the success and the iteration is zero, it is the beginning point of the overfitting process. Also, our algorithm may be developed for multi-class datasets instead of two-class.

References

1. Jang, J.S.R., Sun, C.T., Mizutani, E.: *Neuro-Fuzzy and Soft Computing – a Computational Approach to Learning and Machine Intelligence*. Prentice Hall, Englewood Cliffs (1997)
2. Yu, J.: General C-Means Clustering Model. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27(8), 1197–1211 (2005)
3. Xei, E., Beni, G.: A Validity Measure for Fuzzy Clustering. *IEEE Transaction on Pattern Analysis Machine Intelligence* 13(8), 841–847 (1991)
4. Hekim, M., Orhan, U.: A Validity Measure for a New Hybrid Data Clustering. In: *International Symposium on Innovations in Intelligent Systems and Applications*, Istanbul, pp. 70–74 (2007)
5. Baraldi, A., Blonda, P.: A Survey of Fuzzy Clustering Algorithms for Pattern Recognition, Part I and II. *IEEE Trans. Systems, Man, and Cybernetics, Part B* 29(6), 778–801 (1999)
6. Han, J., Kamber, M.: *Data mining: Concepts and Techniques*. Morgan Kaufmann, San Francisco (2000)
7. Orhan, U., Hekim, M.: Mass Action Based Data Clustering Method and Its Weighted Fuzzification. In: *5th International Conference on Electrical and Electronics Engineering*, pp. 386–390 (2007)
8. Eick, C.F., Zeidat, N., Zhao, Z.: Supervised Clustering – Algorithms and Benefits. In: *Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence*, pp. 774–776 (2004)
9. Basu, S., Bilenko, M., Mooney, R.: Comparing and Unifying Search-based and Similarity-Based Approaches to Semi-Supervised Clustering. In: *20th International Conference on Machine Learning*, Washington, DC, pp. 42–49 (2003)
10. Xing, E.P., Ng, A., Jordan, M., Russell, S.: Distance Metric Learning with Applications to Clustering with Side Information. In: *Advances in Neural Information Processing Systems*, vol. 15, pp. 505–512. MIT Press, Cambridge (2003)
11. Bar-Hillel, A., Hertz, T., Shental, N., Weinshall, D.: Learning Distance Functions Using Equivalence Relations. In: *The Twentieth International Conference on Machine Learning*, Washington (2003)
12. Klein, D., Kamvar, S.D., Manning, C.: From Instance-level Constraints to Space-level Constraints: Making the Most of Prior Knowledge in Data Clustering. In: *The Nineteenth International Conference on Machine Learning*, Sydney, pp. 307–313 (2002)
13. Demiriz, A., Bennett, K.P., Embrechts, M.J.: Semi-supervised Clustering using Genetic Algorithms. *Artificial Neural Networks in Engineering*, 809–814 (1999)
14. Sinkkonen, J., Kaski, S., Nikkila, J.: Discriminative Clustering: Optimal Contingency Tables by Learning Metrics. In: *13th European Conference on Machine Learning*, pp. 418–430 (2002)
15. Slonim, N., Tishby, N.: Agglomerative Information Bottleneck. *Neural Information Processing Systems* (1999)
16. Jin, Y., Dunbrack, R.L.: Assessment of Disorder Predictions in CASP6. *Proteins: Structure, Function, and Bioinformatics* 61, 167–175 (2005)

A Neuro-fuzzy Learning System for Adaptive Swarm Behaviors Dealing with Continuous State Space

Takashi Kuremoto^{1,*}, Masanao Obayashi¹, Kunikazu Kobayashi¹,
Hiroataka Adachi¹, and Kentaro Yoneda²

¹ Graduate School of Science and Engineering, ² Faculty of Science and Engineering,
^{1,2} Yamaguchi University, Tokiwadai 2-16-1, Ube, Yamaguchi, 755-8611, Japan
{wu, m.obayas, koba}@yamaguchi-u.ac.jp

Abstract. Swarm intelligence has brought a new paradise for function optimization, structural optimization, multi-agent systems and other study fields. In our previous work, we proposed a neuro-fuzzy system using reinforcement learning algorithm (actor-critic method with TD error learning algorithm) to acquire optimized swarm behaviors. This paper improves the conventional learning system, which only deals with discrete state space and action space, to solve how a swarm to learn and obtain its adaptive behaviors in the continuous state space. The improved system adopts a new policy function of action which is possible to yield continuous actions corresponding to continuous states. The effectiveness of proposed system is investigated by computer simulations with more kinds of environments for the goal-exploration problem.

Keywords: neuro-fuzzy net; swarm behavior; reinforcement learning; multi-agent system; actor-critic algorithm; goal-exploration problem.

1 Introduction

Animals, such as birds, fishes, ants, bees and so on, organize their individuals to be many kinds of swarms to realize many aims: to defend natural enemies, to find foods, to raise fertility, in one word, the top aim is to keep species. The swarm behavior means intelligence of a species. To explore optimal solution, a famous swarm model named Particle Swarm Optimization (PSO) [1] was proposed by J. Kennedy and R.C. Eberhart and it has brought a new paradise of softcomputing for the field of optimization study in last decade. Dorigo's Ant Colony Optimization (ACO) [2] is another famous model of swarm to solve optimization problems, such as Travelling Salseman Problem (TSP), machine flowshop scheduling problem, etc. All of those swarm models use cooperative behaviors between individuals and global optimal exploration is evaluated iteratively to improve temporary or local solutions. However, these models usually are designed with a prior assumption that particles or individuals have gathered to be a swarm in the environment. The process of how a swarm is formed is

* A part of this work was supported by Grant-in-Aid for Scientific Research (JSPS 18500230, 20500277 and 20500207).

often neglected [3]. To make individuals not only learn to acquire adaptive individual behaviors in the environment but also learn to form a swarm and acquire adaptive swarm behaviors those are necessary to achieve multiple aims for the species, we proposed a reinforcement learning system using a neuro-fuzzy network and actor-critic learning algorithm recently [4].

Our reinforcement learning system for swarm exploration and swarm behaviors formation is designed with a part of fuzzy net and a part of actor-critic learning network. The fuzzy net uses self-multiplication Gaussian membership functions and self-multiplication fuzzy rules to classify states of the environment. Though the fuzzy inference net has powerful ability to classify continuous input and give continuous output [5]-[8], only discrete states and actions were used in our previous system [4]. In the part of actor-critic learning network, the adaptive action is given by actor with a stochastic policy which can be adjusted by connection weights from fuzzy net by learning process. Critic calculates the value of state and the estimated value of next state. After execution of the action, the feedback to the action from the environment, i.e., the reward is used to adjust the stochastic policy too. The reward has negative or positive values is generated on the occasion of obstacle crashing, goal arrival, and near collision or excessive separation from other agents in goal-directed/exploration problems of multi-agent systems. Actor calculates the value of the state and gives a Boltzmann distribution as the stochastic policy for agent. The weights of connections between fuzzy net and actor-critic are modified by temporal difference error (TD error) during online learning.

In this paper, a new policy function is designed to yield continuous action to improve conventional system. Meanwhile, continuous state space is used in the improved system to deal with the real environment. A goal-exploration problem for

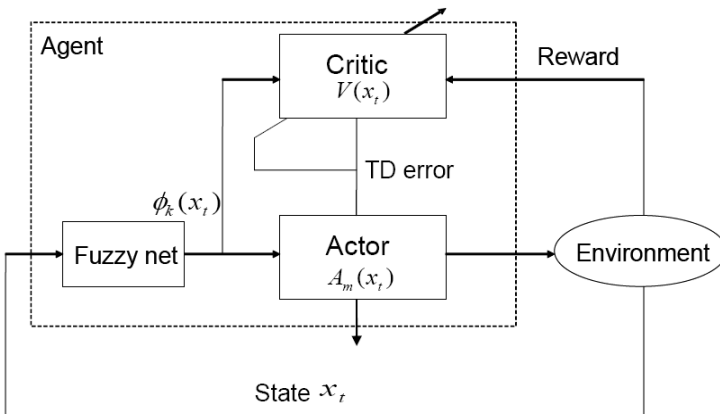


Fig. 1. The structure of a reinforcement learning system for swarm behaviors is shown. The system for each agent has a part of fuzzy net and a part of actor-critic learning network. The reward is given by not only the environment but also other agents.

multiple agents was applied in simulation experiments using different environments, and the efficiency of improved system was confirmed by the simulation results.

2 A Neuro-fuzzy Learning System

The architecture of our neuro-fuzzy reinforcement learning system for swarm behaviors is shown in Fig. 1. An agent observes states x_t from the environment and classifies the inputs into k classes $\phi_k(x_t)$ by Fuzzy net. The agent executes an action which may change the state by its policy function given by Actor. Critic receives rewards from the environment or external evaluators and connects with Fuzzy net by synapse (weight). The value of state is calculated by state-value function $V(x_t)$, and its temporal change results TD error which is used to adjust policy function through action-value function $A(x_t)$ given by Critic. Each agent has the same architecture and when swarm behaviors are evaluated with positive rewards, agents learn to form swarm and act more efficiently than individual situation.

2.1 Fuzzy Net

For an n -dimension input state space $\mathbf{x}(x_1(t), x_2(t), \dots, x_n(t))$, a fuzzy inference net is designed with a hidden layer which units are RBF-like fuzzy membership functions $B_i^k(x_i(t))$ to classify input states, and fuzzy rules are generated by multiplying their corresponding membership functions as same as in [4]. The number of membership functions and rules of fuzzy net are important for a fuzzy inference system. We proposed a self-organized fuzzy neural network (SOFNN) which constructed adaptive membership functions and rules using training data and thresholds previously [7] and [8]. Wang, Cheng, and Yi proposed a structure learning algorithm for adding and merging units using TD error distribution recently [5]. The self-multiplication algorithm to decide the size of fuzzy net in [4] is also used here.

2.2 Actor-Critic Learning Network for Continuous Action

The weighted outputs of fuzzy net are used to calculate the value of states and actions which belong to Actor and Critic respectively [4].

Now, let actor function $A_j(x(t)) \in R^m$ denotes the j th action (behavior) selected by agent according to a stochastic policy Eq. (3), where $j = 1, 2, \dots, m$.

$$P(a_t = a_j | \mathbf{x}(t)) = \frac{\exp(A_j(\mathbf{x}(t))/T)}{\sum_m \exp(A_m(\mathbf{x}(t))/T)} \tag{1}$$

Here T is the temperature of Boltzmann distribution. Higher T causes more active exploration, and lower temperature causes more greedy action to the goal.

In our previous study, however, only discrete actions were permitted for the simulation of goal-exploration problem. Agents moved 1 grid by 1 step, one and only one of 4 candidature directions is limited for the selection of action. We propose use more

information of m -dimension here by a linear combination of plural actions. In fact, policy function Eq. (1) decides one action which has highest probability in m candidates conventionally. The stochastic selection of traditional action policy usually neglects $m-1$ actions for their lower probabilities. Now, let z enough high probabilities' linear combination $P_t(a_t^{new} = \sum_{i=1}^z a_i | x(t))$ express the probability of a new action a_t^{new} :

$$P_t(a_t^{new} = \sum_{i=1}^z a_i | x(t)) = \sum_{j=1}^z P_t(a_i = a_j | A_m(x(t))) \leq 1.0 \tag{2}$$

where $n \leq m$. So an action set (a_1, a_2, \dots, a_z) becomes a dominant candidate. The new action a_t^{new} may be generated by the linear combination in the dominant action vector space as shown as Eq. (3). The value $A_{new}(x(t))$ is given by Eq. (6) which concerns with all values of dominant actions.

$$\vec{a}_t^{new} = k_1 \vec{a}_1 + k_2 \vec{a}_2 + \dots + k_z \vec{a}_z \tag{3}$$

$$A_{new}(\vec{a}_t^{new} | x(t)) = \|\vec{a}_t^{new}\|_z = \left(\sum_{i=1}^z k_i^z \right)^{\frac{1}{z}} \tag{4}$$

where $z \leq m$. Coefficients k_1, k_2, \dots, k_z are scalar values given by action value function Eq. (4) and actions a_1, a_2, \dots, a_z become to the basis of action vector space, $z = 2$ when the action vector is in Euclidean space.

Critic owns its value and temporal difference error (TD error) by reward from the environment resulted by actor's action. TD error $\epsilon_{TD}(x(t))$ and the learning algorithm are as same as which in [4].

2.3 Swarm Formation and Adaptive Swarm Behaviors

To form swarm and acquire swarm intelligence by reinforcement learning process, agents obtain reward not only when they arrive at the goal and crash to obstacles or themselves, but also when a swarm is formed. Let $D(x^p(t), x^q(t))$ express the Euclidean distance between agent p and agent q , then swarm reward can be given by judging whether $D(x^p(t), x^q(t))$ is suitable for individual's gathering [4].

3 Simulation Experiments

3.1 Problem Description

A goal-exploration problem was used to investigate performance of the improved learning system. Two squares with the same size of 10x10 were used as exploration spaces which were smaller than the exploration square in simulation of discrete system [4]. There were walls on the four sides, no obstacle in one environment and more

walls in the square in another environment, and goal area was fixed as shown in Fig. 2. Each agent observed its position and position of others, and moved 1.0 length per step toward an arbitrary direction while only 4 probable directions was allowed in the conventional system simulations, i.e., up, down, left and right. The method to decide the direction of one time movement was to choose 2 orthogonal directions whose value of action function were higher than others. Agents did not have any information of the goal position before they arrived at it.

Two kinds of simulation experiments were performed. Simulation I: multiple agents learn to search the goal as fast as they can individually (*individual learning*); Simulation II: multiple agents learn to search the goal as fast as the can with swarm reward in the same environment respectively (*swarm learning*). Conventional system used discrete action and improved system used continuous action were executed in the two environments. The values of parameters used in these simulations were shown in Table I. The number of learning iterations for one episode (or one trial: an exploration from the start to the goal) was set to be not over 2,000 in all kinds of simulations.

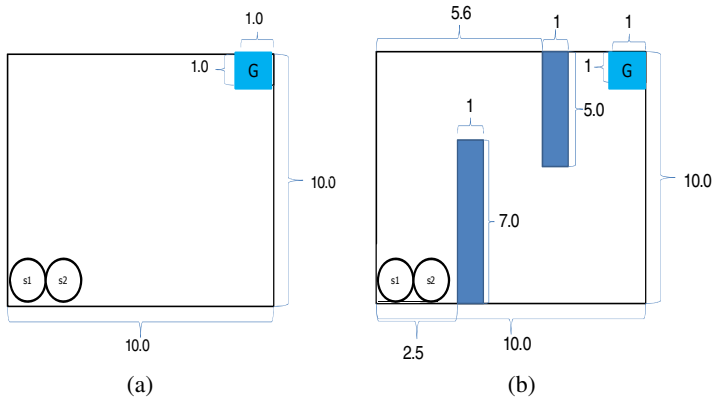


Fig. 2. Simulation environments for two agents goal-directed exploration experiment. Each exploration area was set to be a square with a size of 10x10. Agents s_1, s_2 started from left-down corners. Square G on the right-up corner was set as the goal area. There was no any obstacle in environment (a), and two obstacles existed in maze-like environment (b). The length of agent step was 1.0, and direction of the movement was arbitrary angle given with real number.

3.2 Simulation Results

Simulation I (i): *Individual learning* in non-obstacle environment (Fig. 3). Both systems showed enough learning ability of the neuro-fuzzy system dealing with goal-exploration problem of multiple agents. Simulation I (ii): *Swarm learning* in non-obstacle environment (Fig. 4). Swarm formation could be observed comparing with results of Simulation I (i). Simulation II (i): *Individual learning* in maze-like environment. Fig. 5 (left) and (right) showed the learning results of conventional system and improved system respectively. Simulation II (ii): *Swarm learning* in maze-like environment (Fig. 6). Swarm formation could be observed comparing with results of

Simulation II (i). In this kind of environment, *swarm learning* showed more efficiently to acquire shorter routes comparing with *individual learning*. This result was not confirmed in our previous work [4]. Fig. 7 shows how the number of exploration steps reduced with learning algorithm iterations using improved system. Fig. 7 (a) shows the case of Simulation I (i) and (ii) which used non-obstacle environment. There was no obvious difference between *individual learning* method and *swarm learning* method. Fig. 7 (b) shows the case of Simulation II (i) and (ii) which used maze-like environment. Comparing the gradient of learning curves in Fig. 7 (a) and (b), it is suggested that *swarm learning* method gave a faster convergence than *individual learning* in the exploration of complicate environment.

Table 1. Parameters used in the simulations

Description	Symbol	Quantity
Dimension of input vectors	n	2
The number of actions	m	2
Standard deviation of membership	σ_i^k	0.1
Threshold of fitness	F	0.4
Initial weight between rules and critic	v_k	1.0
Initial weight between rules and actor	w_{kj}	0.25
TD learning coefficient for critic	β_v	0.3
TD learning coefficient for actor	β_w	0.3
Discount of TD error	γ	0.9
Temperature of Boltzmann distribution	\mathcal{T}	0.1
Reward for goal arrived	r_{goal}	100.0
Reward for wall or agent crashed	r_o	-10.0
Reward for corner crashed	r_c	-20.0
Reward for swarm formed	r_{swarm}	1.0
Reward for swarm unformed	$r_{a-swarm}$	-1dis
Minimum distance between agents	min_dis	1.5
Maximum distance between agents	max_dis	3.0

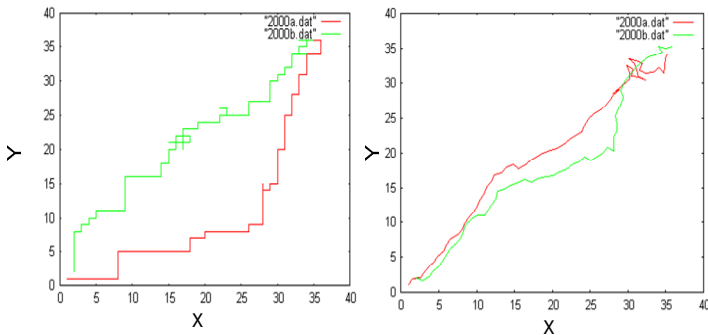


Fig. 3. *Individual learning* results in the environment Fig. 2 (a) were shown: **(left)** used discrete input and output; **(right)** used continuous input and output

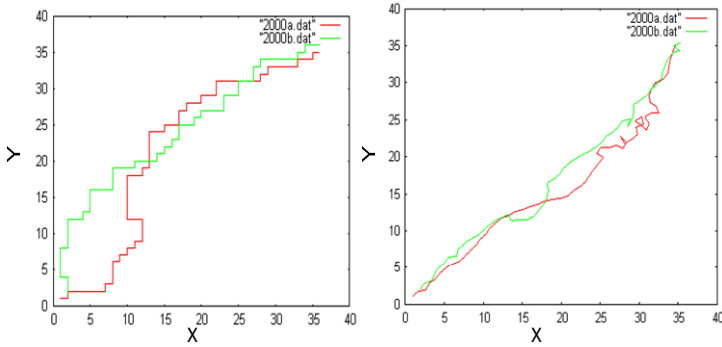


Fig. 4. *Swarm learning* results in the environment of Fig. 2 (a) were shown: **(left)** used discrete input and output; **(right)** used continuous input and output. Swarm can be confirmed comparing with Fig. 3 (left) and (right) respectively.

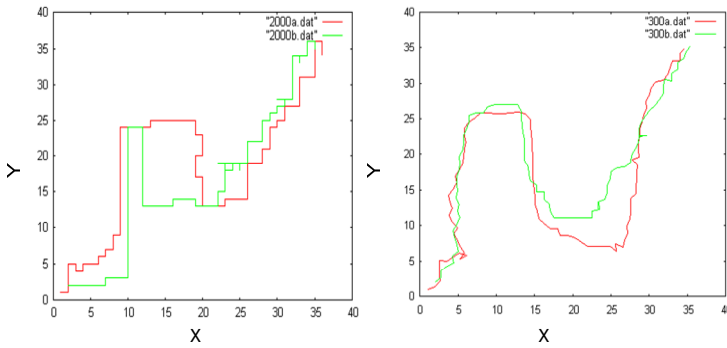


Fig. 5. *Individual learning* results in the environment Fig. 2 (b) were shown: **(left)** used discrete input and output; **(right)** used continuous input and output

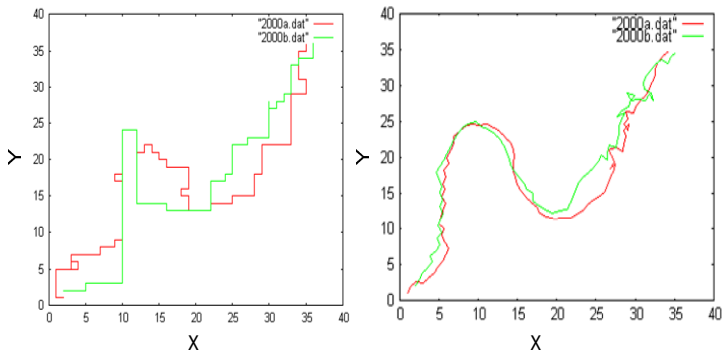


Fig. 6. *Swarm learning* results in the environment Fig. 2 (b) were shown: **(left)** used discrete input and output; **(right)** used continuous input and output. Swarm can be confirmed comparing with Fig.5 respectively.

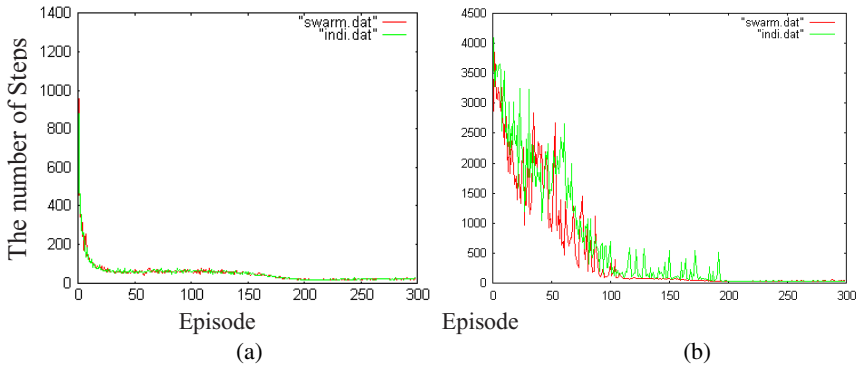


Fig. 7. Learning process: the exploration steps reduced by learning iterations using either *individual learning* or *swarm learning* method. (a) shows the case of non-obstacle environment; (b) shows the case of maze-like environment. Furthermore, 3 and 4 agents were also tested but the similar convergences to the above results were observed.

4 Conclusions

A neuro-fuzzy reinforcement learning system for optimal behaviors acquisition and swarm formation was improved to deal with continuous state and continuous action in this paper. The new idea proposed here is to compose a new continuous action by linear combination of dominant actions. Two kinds of goal-exploration environments, i.e., non-obstacle environment and maze-like environment, were used in the simulation experiments. The efficiency of the improved system was confirmed comparing with conventional one. The results showed new system also has enough learning ability of swarm formation and adaptive behavior acquirement. The future work of this study is expected to avoid using coordinate information but perspective information in real environment.

References

1. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: Proc. IEEE Int. Conf. Neural Networks, vol. 4, pp. 1924–1948 (1995)
2. Dorigo, M.: Artificial life: The swarm intelligence approach. In: Congress on Evolutionary Computing (1999)
3. Iima, H., Yasuaki, K.: Swarm reinforcement learning algorithm based on exchanging information among agents. Trans. SICE (in Japanese) 42(11), 1244–1251 (2006)
4. Kuremoto, T., Obayashi, M., Kobayashi, K., Adachi, H., Yoneda, K.: A Reinforcement Learning System for Swarm Behaviors. In: Proc. 2008 IEEE World Congress on Computational Intelligence (WCCI / IJCNN 2008), pp. 3710–3715 (2008)
5. Wang, X.S., Cheng, Y.H., Yi, J.Q.: A fuzzy Actor–Critic reinforcement learning network. Information Sciences 177, 3764–3781 (2007)

6. Jouffe, L.: Fuzzy inference system learning by reinforcement learning. *IEEE Trans. Sys., Man and Cybernetics* 28(3), 338–355 (1998)
7. Kuremoto, T., Obayashi, M., Yamamoto, A., Kobayashi, K.: Predicting chaotic time series by reinforcement learning. In: *Proc. 2nd Int. Conf. CIRAS* (2003)
8. Kuremoto, T., Obayashi, M., Kobayashi, K.: Nonlinear prediction by reinforcement learning. In: Huang, D.-S., Zhang, X.-P., Huang, G.-B. (eds.) *ICIC 2005. LNCS*, vol. 3644, pp. 1085–1094. Springer, Heidelberg (2005)

A Reinforcement Learning Automata Optimization Approach for Optimum Tuning of PID Controller in AVR System

Mohammad Kashki¹, Youssef Lotfy Abdel-Magid², and Mohammad Ali Abido³

¹ Electrical Eng. Dept. of Shahid Bahonar University, Kerman, Iran

² Electrical Engineering Program, The Petroleum Institute, Abu Dhabi, UAE

³ Dept. of Electrical Eng., King Fahd University of Petroleum & Minerals, Dhahran, SA

Abstract. In this paper, an efficient optimization method based on reinforcement learning automata (RLA) for optimum parameters setting of conventional proportional-integral-derivative (PID) controller for AVR system of power synchronous generator is proposed. The proposed method is Continuous Action Reinforcement Learning Automata (CARLA) which is able to explore and learn to improve control performance without the knowledge of the analytical system model. This paper demonstrates the full details of the CARLA technique and compares its performance with Particle Swarm Optimization (PSO) and Genetic Algorithms (GA) as two famous evolutionary optimization methods. The simulation results show the superior efficiency and performance of the proposed method in regard to other ones.

Keywords: reinforcement learning automata; CARLA; PID; evolutionary computations.

1 Introduction

The PID controller is the most frequently used control element in the industrial world in comparison to other controllers such as adaptive controllers, artificial neural network based controllers, fuzzy and neuro-fuzzy controllers. It is estimated that, at least, 90% of the controllers employed in the industry are PIDs or its variants [1]. The popularity of the PID controller is attributed to its simple structure, high reliability, and robust performance in a wide range of operating conditions. Despite all of the PID controller good features, unfortunately, its appropriate gain tuning is still a problem in many practical industrial applications because of high order, time delay and nonlinearity of the plants [2]. The normal tuning method in many applications is carried out using the classical tuning rules proposed by Ziegler-Nichols [3], which in general, does not yield optimal or near-optimal behavior in many industrial plants and just cannot be counted as a feasible solution.

In recent years, many heuristic methods for the optimum tuning of PID parameters such as genetic algorithms (GA) and simulated annealing (SA) have been proposed with noticeable success in solving complex optimization problems [4-9]. Recently, a modern heuristic algorithm called *Particle Swarm Optimization* was proposed by Kennedy and

Eberhart which is developed through simulation of a simplified social system and have been found to be robust in solving nonlinear optimization problems [10].

In this paper a reinforcement learning automata based method called *Continuous Action Reinforcement Learning Automata* (CARLA) which was first introduced by Howell, Frost, Gordon and Wu [14] is used for the optimum tuning of the PID controller of a synchronous generator, and its performance is compared with PSO and GA based PID controllers which are completely investigated by Gaing [13]. The CARLA operates through interaction with a random or unknown environment by selecting actions in a stochastic trial and error process. The CARLA method has been successfully applied to different kind of optimization problems [14-15].

The generator excitation system maintains generator voltage and controls the reactive power flow using an automatic voltage regulator (AVR) [16]. The role of an AVR is to hold the terminal voltage magnitude of a synchronous generator at a specified level. Hence, the stability of the AVR system would seriously affect the security of the power system.

2 PID Controller

The PID controller is composed of three main components: proportional, integral and derivative. Fig. 1 shows this structure and components.

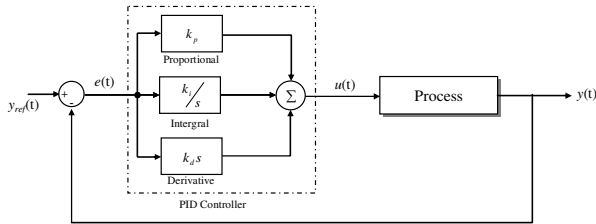


Fig. 1. PID structure

Where $y_{ref}(t)$ is the reference output, $e(t)$ is the error, $u(t)$ is the control signal, and $y(t)$ is the output. Each of the PID controller components has its own specific effect on the controller performance: the Proportional component has the effect of increasing the loop gain to make the system less sensitive to disturbances, the integral component is used principally to eliminate steady-state errors, and the derivative action helps to improve closed loop stability [14]. The gain parameters k_p, k_i, k_d are thus chosen to meet prescribed performance criteria, classically specified in terms of rise and settling times, overshoot and steady state error, following a step change in the reference output signal. The transfer function of PID controller can be expressed by Eq. (1).

$$u(t) = k_p e(t) + k_i \int_0^t e(t) dt + k_d \frac{de(t)}{dt} \tag{1}$$

Thus, the decision variables involved in the optimization problem are the gain parameters: k_p, k_i, k_d

3 Automatic Voltage Regulator

The role of an AVR is to hold the terminal voltage magnitude of a synchronous generator at a specified level. A simple AVR system is composed of four main components, namely the amplifier, the exciter, the generator, and the sensor. For comparison with PSO and GA based PID controllers, in this paper, a linearized model of the AVR is considered which takes into account the major time constant of the AVR components and ignores nonlinearities [11], as shown in Fig. 2.

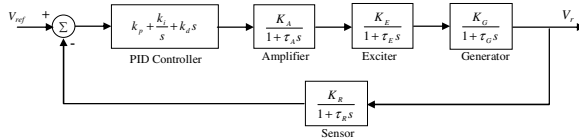


Fig. 2. Block diagram of linearized AVR with PID controller

Where V_{ref} is the reference voltage and V_r is the excitation voltage. Table 1 summarizes the typical range of linearized model parameters.

Table 1. Typical range of AVR linearized model parameters

AVR component	Parameter	Typical range
Amplifier	K_A	[10,400]
	τ_A	[0.02,0.1]
Exciter	K_E	[10,400]
	τ_E	[0.5,1]
Generator	K_G	[0.7,1]
	τ_G	[1,2]
Sensor	K_R	[1,2]
	τ_R	[0.001,0.06]

4 Continuous Action Reinforcement Learning Automata

In The CARLA optimization method, a continuous probability density function (CPDF) is associated with each decision variable, and through modification of these CPDFs over sufficient number of iterations the optimal value of the decision variables will be determined. The modification process in each iteration is due to reinforcement signal corresponding to a predefined cost function. Fig. 3 shows the diagram of CARLA. The optimization process of the decision variables runs in parallel due to minimization of a predefined cost function and the only interconnection between them is through the environment, i.e. PID controller and AVR system, and via shared performance evaluation function. The probability density functions are the basis of random action selection. The CPDFs are initially defined uniformly as Eq. 2.

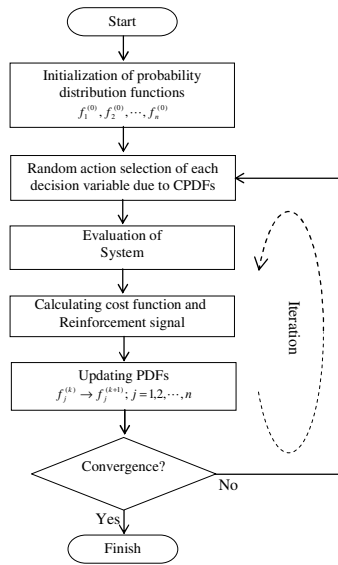


Fig. 3. Diagram of CARLA optimization method

$$f_i^{(0)}(x_i) = \begin{cases} \frac{1}{x_{i,max} - x_{i,min}} & x_i \in [x_{i,min}, x_{i,max}] \\ 0 & otherwise \end{cases} \quad i = 1, 2, \dots, n \quad (2)$$

Where, n is the number of decision variables, x_1, x_2, \dots, x_n are the decision variables and f_1, f_2, \dots, f_n are the corresponding CPDFs. Actions, i.e. selected values of decision variables in each iteration, is selected by Eq. 3.

$$\int_0^{x_i} f_i^{(k)}(x_i) = z_i(k) \quad i = 1, 2, \dots, n \quad (3)$$

Where k is the iteration number and z varies randomly in the range $[0,1]$. When all actions were selected, the PID controller will be constructed and applied to the plant for a suitable time. In this paper this step is done by computer simulation, but in practical applications it can be a real-time evaluation. After this evaluation, a scalar cost value is calculated according to a predefined cost function. Normally IAE , ISE , and $ITSE$ functions are selected as a cost function, but every one of these criteria has some disadvantages [13]. Moreover, for accurate comparison with PSO-PID and GA-PID the same cost function is selected as expressed in Eq. 4.

$$J^{(k)} = (1 - e^{-\lambda}).(M_p + E_{ss}) + e^{-\lambda}.(t_s - t_r) \quad (4)$$

Which $J^{(k)}$ is the cost function of the k th iteration, λ is the weighting factor and set between 0.8 to 1.5, M_p is the overshoot of the output signal, E_{ss} is the steady state error, and t_s, t_r are the settling and rising time respectively. It is reiterated that, the CARLA algorithm does not require the knowledge of the system dynamics, but the

designer should be aware of system behaviors in order to define an appropriate cost function.

The performance evaluation and consequent modification of the CPDFs are carried out by the reinforcement signal which is defined by Eq. 5.

$$\beta^{(k)} = \min \left\{ 1, \max \left\{ 0, \frac{J_{mean} - J^{(k)}}{J_{mean} - J_{min}} \right\} \right\} \tag{5}$$

Where $\beta^{(k)}$ is the reinforcement signal in the k th iteration and J_{mean} , J_{min} are the mean and minimum values of previous iterations cost value, respectively. This definition of the reinforcement signal performs a reward/inaction rule in CPDFs modification. In the other word, if current selected actions are less that mean value of previous cost, i.e. $\beta=0$, then no modification of CPDFs must be performed (inaction) and, if selected action lead to cost value less than minimum of previous cost, i.e. $\beta=1$, then maximum reinforcement will be done (reward). The CPDFs modification is done by Eq. 6.

$$f_i^{(k+1)}(x) = \begin{cases} \alpha_i^{(k)} (f_i^{(k)}(x) + \beta^{(k)} H_i(x_i, \tilde{x}_i)) & x_i \in [x_{i,min}, x_{i,max}] \\ 0 & otherwise \end{cases} \tag{6}$$

Where, $H_i(x_i, \tilde{x}_i)$ is a symmetrical Gaussian function that is centralized at chosen actions \tilde{x}_i and defined as Eq. 7.

$$H_i(x, \tilde{x}_i) = \frac{g_h}{x_{i,max} - x_{i,min}} \exp \left(- \frac{(x - \tilde{x}_i)^2}{2(g_w(x_{i,max} - x_{i,min}))^2} \right) \tag{7}$$

Where g_h , g_w are the normalized height and width of Gaussian function, respectively, and they determine the speed and resolution of learning. The Gaussian function is used for changing the probability of selected actions as while as their neighbor actions. The parameter α_i is the distribution normalization factor in the $(k+1)$ th iteration and is defined as Eq. 8.

$$\alpha_i^{(k)} = \frac{1}{\int_{x_{i,min}}^{x_{i,max}} f_i^{(k)}(x_i) + \beta^{(k)} H(x_i, \tilde{x}_i) dx_i} \tag{8}$$

At the end, the convergence criterion of the algorithm determines if the algorithm should stop or not. This criterion can be a specified number of iteration or stand stillness of selected actions, etc. After the algorithm is halted, it is expected that the CPDFs are maximized at corresponding decision variable optimal value.

5 Computer Simulations and Results

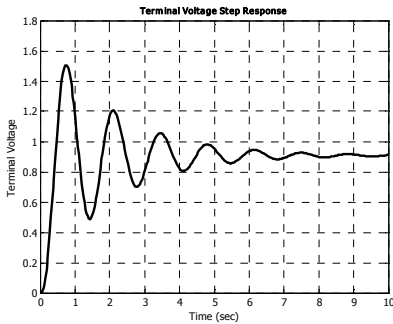
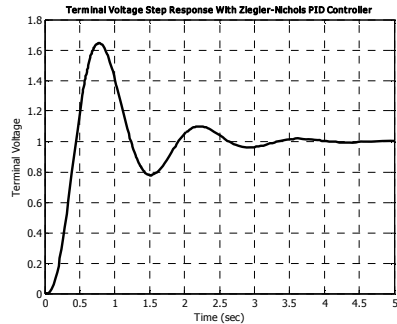
To verify the efficiency of designed CARLA-PID controller a practical high-order AVR system is used in computer simulation. The simulation results are carried out in Matlab® and Simulink® environments. Furthermore, the performance of proposed CARLA-PID is compared with PSO-PID and GA-PID controller with same performance index criterion. Table 2 summarizes the parameters of AVR system.

Table 2. The test AVR model parameters

AVR component	Parameter	Value
Amplifier	K_A	10
	τ_A	0.1
Exciter	K_E	1
	τ_E	0.4
Generator	K_G	1
	τ_G	1
Sensor	K_R	1
	τ_R	0.01

To emphasize the necessity of using PID controller in this AVR system, the step response of the terminal voltage without PID controller was simulated as shown in Fig. 4.

As can be seen, the overshoot and steady state error are about 50.51% and 8.81%, respectively. The Ziegler-Nichols method for designing the PID controller will result in the gain parameter: $k_p=1.0228$, $k_i=1.8423$, and $k_d=0.1357$. The terminal voltage response with of the Ziegler-Nichols PID controller is shown in Fig. 5.

**Fig. 4.** Terminal voltage step response without PID controller**Fig. 5.** Terminal voltage step response with Ziegler-Nichols PID controller

Although, the Ziegler-Nichols-PID controller succeeded into eliminating the steady state error, the overshoot value is still high, i.e. $M_p=64.42\%$.

5.1 Optimal PID Controllers

The performance evaluation of the optimal PID controllers, namely, the CARLA-PID, the PSO-PID and the GA-PID, is performed for two different values of weighting factor in the cost function, i.e. $\lambda=1.5$ and $\lambda=1$. The parameters of the CARLA-PID algorithm are as follows:

- the decision variables are: $x_1=k_p$, $x_2=k_i$ and $x_3=k_d$ ($n=3$);
- upper and lower bounds of decision variables are as $0 \leq k_p \leq 1.5$, $0 \leq k_i \leq 1$, $0 \leq k_d \leq 1$;
- height and width of Gaussian function are $g_h=0.7$ and $g_w=0.03$.

Fig. 6 shows the CPDFs variations toward algorithm iterations for $\lambda=1.5$,

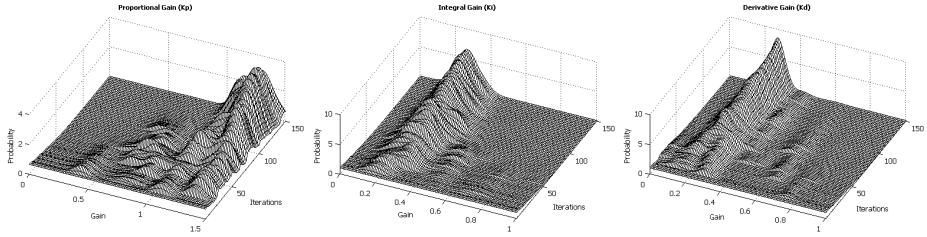


Fig. 6. CPDFs variation of CARLA-PID gain parameters

The trend of convergence is shown in Fig. 7 which shows that the CARLA algorithm converges in about 89 iterations for $\lambda=1.5$ and 120 iterations for $\lambda=1$, while, PSO and GA converge in less than 30 iterations.

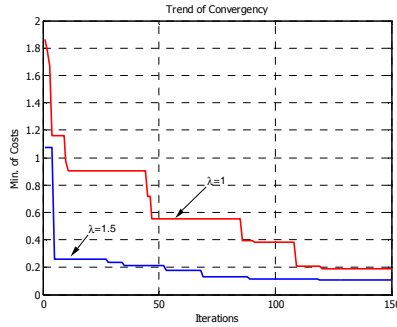


Fig. 7. Trend of convergence of CARLA-PID for different values of λ

The parameters of PSO-PID controller are as follows:

- the member of each individual is k_p , k_i and k_d ;
- population size =50
- the limit of change in velocity for each member is half of the corresponding gain parameter maximum value as same as in CARLA-PID

And finally, the parameters of GA-PID controller with Elitism scheme [5,6] are as follows:

- the member of each individual is k_p , k_i and k_d ;
- population size =50
- crossover rate $P_c = 0.6$;
- mutation rate $P_m = 0.01$;

5.2 Comparison of Optimal PID Controllers

Table 3 summarizes the best solution of the optimum CARLA-PID, PSO-PID and GA-PID controllers gains followed by the cost function value. Fig. 8 shows the

Table 3. Comparison of optimal PID controllers

Weight	$\lambda=1.5$			$\lambda=1$		
Parameters	PSO-PID	GA-PID	CARLA PID	PSO-PID	GA-PID	CARLA PID
k_p	0.6476	0.8935	1.2191	0.6570	0.8663	1.0184
k_i	0.5216	0.6458	0.2943	0.5390	0.7531	0.2809
k_d	0.2375	0.4014	0.2742	0.2458	0.3365	0.2308
M_p (%)	14.91	14.73	1.2	15.38	16.58	1.93
E_{ss}	0	0	0	0	0	0
t_s (s)	3.8929	4.3684	0.5961	3.8540	3.7042	0.6462
t_r (s)	0.1972	0.1787	0.1530	0.1934	0.1738	0.1689
Cost Value	0.9408	1.0499	0.1084	1.4442	1.4056	0.1879

efficiency of each optimal PID controller used in AVR system corresponding to different values of weighting factor in cost function.

As can be seen by results, the CARLA-PID has more efficiency and performance in compare with two other PIDs. Moreover, Fig. 9 shows the control signal of optimal PID controllers.

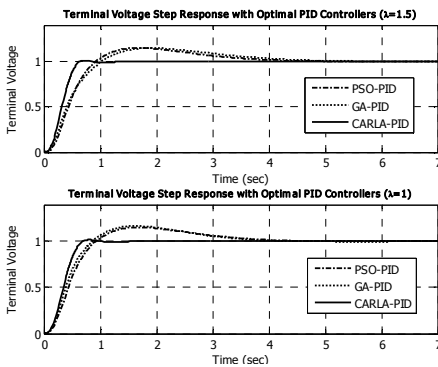


Fig. 8. Terminal voltage step response with optimal PID controllers

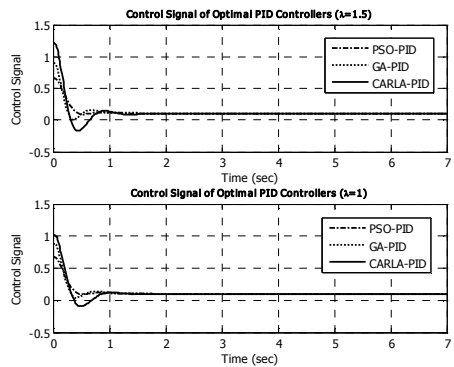


Fig. 9. Control signal of optimal PID controllers

6 Conclusion

An efficient design method using *Continuous Action Reinforcement Learning Automata* for the optimal tuning of PID conventional controller used in AVR system of synchronous generator is proposed. The method does not require the knowledge of the dynamics and equations of plant. In addition, the performance of the proposed method is compared with *Particle Swarm Optimization* and *Genetic Algorithm* methods. As indicated by simulation results, the speed of convergence of the proposed method is less than other methods but its performance is quite superior to the GA and the PSO . Therefore, the proposed method can overcome some of the

shortcomings of other optimization techniques and can be widely used in more complex optimization problems.

References

1. Santos, M., de la Cruz, J.M.: Between Fuzzy-PID and PID-Conventional Controllers: a Good Choice, 123–127 (1996)
2. Visioli, A.: Tuning Of PID Controllers with Fuzzy Logic. *Instrument Electrical Engineering Control Theory Application* 148(1), 1–8 (2001)
3. Ziegler, J., Nichols, G.: Optimum Setting for Automatic Controllers. *Transactions of ASME* 64(1), 759–768 (1942)
4. Seng, T.L., Khalid, M.B., Yusof, R.: Tuning of a Neuro-Fuzzy Controller by Genetic Algorithm. *IEEE Transaction on System, Man, Cybernetic* 29, 226–236 (1999)
5. Kawabe, T., Tagami, T.: A Real Coded Genetic Algorithm for Matrix Inequality Design Approach of Robust PID Controller with Two Degrees of Freedom. In: 12th IEEE International Symposium on Intelligent Control, Istanbul, Turkey, pp. 119–124 (1997)
6. Krohling, R.A., Jaschek, H., Rey, J.P.: Designing PI/PID Controller For A Motion Control System Based on Genetic Algorithm. In: 12th IEEE International Symposium on Intelligent Control, Istanbul, Turkey, pp. 125–130 (1997)
7. Kwok, D.P., Sheng, F.: Genetic Algorithm and Simulated Annealing for Optimal Robot Arm PID Control. In: *IEEE Conference of Evolutionary Computation*, Orlando, FL, pp. 707–713 (1994)
8. Ota, T., Omatu, S.: Tuning of The PID Control Gains By GA. In: *IEEE Conference of Emerging Technology Factory Automation*, Kauai, HI, pp. 272–274 (1996)
9. Jones, A.H., Oliveira, P.B.D.: Genetic Auto-Tuning of PID Controllers. In: *Instrument Electrical Engineering Conference on Genetic Algorithm and Engineering System Innovations Application*, pp. 141–145 (1995)
10. Kennedy, J., Eberhart, R.: Particle Swarm Optimization. In: *IEEE International Conference Neural Networks*, Perth, Australia, pp. 1942–1948 (1995)
11. Yoshida, H., Kawata, K., Fukuyama, Y.: A Particle Swarm Optimization for Reactive Power and Voltage Control Considering Voltage Security Assessment. *IEEE Transaction on Power Systems* 15, 1232–1239 (2000)
12. Howell, M.N., Frost, G.P., Gordon, T.J., Wu, Q.H.: Continuous Action Reinforcement Learning Applied to Vehicle Suspension Control. *Mechatronics* 7(3), 263–276 (1997)
13. Gaing, Z.L.: A Particle Swarm Optimization Approach for Optimum Design of PID Controller in AVR System. *IEEE Transactions On Energy Conversion* 19(2), 384–391 (2004)
14. Howell, M.N., Best, M.C.: On-Line PID Tuning for Engine Idle-Speed Control Using Continuous Action Reinforcement Learning Automata. *Control Engineering Practice* 8, 147–154 (2000)
15. Howell, M.N., Gordon, T.J.: Continuous Action Reinforcement Learning Automata and Their Application to Adaptive Digital Filter Design. In: *Engineering Applications of Artificial Intelligence*, pp. 549–562. Elsevier Science Ltd., Amsterdam (2001)
16. Saadat, H.: *Power System Analysis*. McGraw-Hill, New York (1999)

An Alternative to Center-Based Clustering Algorithm Via Statistical Learning Analysis

Rui Nian^{1,2}, Guangrong Ji², and Michel Verleysen¹

¹ Machine Learning Group, DICE, Université catholique de Louvain, Place du Levant,
3-B-1348 Louvain-la-Neuve, Belgium

² College of Information Science and Engineering, Ocean University of China, Qingdao,
China, 266003

nianrui_80@163.com, grji@mail.ouc.edu.cn,
michel.verleysen@uclouvain.be

Abstract. This paper presents an alternative for center-based clustering algorithms, in particular the k-means algorithm, via statistical learning analysis. The essence of statistical learning principle, i.e., both the empirical risk and structural assessment, is taken into particular consideration for the clustering algorithm so as to derive and develop the relevant minimization mathematical criterion with automatic parameter learning and model selection in parallel. The proposed algorithm roughly decides on the number of clusters, by earning activation for the winners and assigning penalty for the rivals, so that the most competitive center wins for possible prediction and the extra ones are driven far away when starting the algorithm from a too large number of clusters without any prior knowledge. Simulation experiments prove the feasibility of the algorithm and show good performances of the double learning tasks during clustering.

1 Introduction

Clustering, an unsupervised learning process to pursue natural groups among unlabeled data, is one of the most important tasks in Intelligent Computing and Machine Learning. Typically, there is still no perfect solution for the generation and evaluation of clusters. Clustering algorithms proposed in the literature try to seek the minimization of certain mathematical criterion as good as possible [1-4].

In practice, center-based clustering has shown its generality, maneuverability and effectiveness for many applications [1-3]. Usually assuming that each cluster adheres to a unimodal distribution, center-based clustering algorithms try to make each center describe the truth in a single cluster drawn from one mode. However, there still remain some problems not yet completely solved in center-based clustering. First, the best value for the number of clusters is not always clear; it is usually required to specify the number of clusters beforehand in most cases, which is often an ad hoc decision based on prior knowledge, assumptions, and practical experience, and becomes more difficult in a high dimensional space. Second, in mixture distribution cases, it is still a NP hard problem to select and partition data into approximately original clusters which capture and reflect the natural attributes among data.

In virtue of statistical tools, problems existing in clustering could be to some extent discovered, analyzed and solved. Classical statistics typically focuses on sufficient statistic cases, while statistical learning theory is a machine learning principle to explore the inherent distribution, dependence structure, and generalization ability as good as possible from a finite sample size [5, 6]. Vapnik first put forward statistical learning theory for model complexity based on a minimal capacity measure - VC dimension confidence [5]. Lei Xu also proposed a general statistical learning framework, Bayesian Ying-Yang harmony learning theory, for simultaneous parameter learning and model selection [6].

In this paper, with the help of the general statistical learning principle from a direct perspective, we heuristically explore an alternative conceptually equivalent to the previous work [4, 7], for center-based clustering algorithms, in particular for k-means algorithm, by taking both the empirical risk and structural assessment into consideration. We derive the relevant minimization mathematical criterion with joint parameter learning and model selection, try to seek one solution to learn about both the range and the number of the clusters in mixture distribution cases simultaneously, and accomplish the double learning tasks during clustering procedure. Simulation experiments prove the feasibility of the algorithm and show good performances for both the clustering itself and the estimation on the number of clusters.

2 Center-Based Clustering Algorithms

Center-based clustering algorithms consider that each cluster follows a unimodal distribution and attempt to seek centers from natural clusters [3]. Given an input data set $X = \{x_t\}_{t=1}^N$ drawn from K^* true clusters, the task of center-based clustering is to partition X into K categories, each being represented by an inner center y_ℓ in the representation domain $Y = \{y_\ell\}_{\ell=1}^K$ in a machine learning system. The k-means algorithm is one of the most popular center-based clustering algorithms. The basic idea of k-means algorithm is to partition data into clusters with the objective that tries to achieve the minimization of the total intra-cluster variance, or, the Mean Square Error (MSE) function [1]. Similar to the k-means algorithm, the Expectation-Maximization (EM) algorithm for mixtures of Gaussians is another widely studied method in center-based clustering algorithms, which maximizes the likelihood estimation in probabilistic models that depends on unobserved latent variables [2].

In this paper, for the center-based clustering algorithms, we lay emphasis on the k-means algorithm and explore some improvements. In general, the membership for k-means algorithm is:

$$M(y_\ell | x_t) = \begin{cases} 1 & \text{if } \ell = \ell_t \\ 0 & \text{otherwise} \end{cases}, \quad (1)$$

$$\ell_t = \arg \min_j d^2(x_t, y_j)$$

where $d(x_i, y_j)$ is the similarity measure between x_i and y_j , and each input is assigned to the nearest cluster label ℓ_i . The objective function for optimization in the k-means algorithm is the Mean Square Error function as follows:

$$E_{MSE} = \frac{1}{N} \sum_{\ell=1}^K \sum_{i=1}^N M(y_\ell | x_i) d^2(x_i, y_\ell) = \frac{1}{N} \sum_{i=1}^N d^2(x_i, y_{\ell_i}) . \tag{2}$$

Here Y is obtained by minimizing this objective function $\min_Y E_{MSE}$. Only one winner ℓ_i is activated and its corresponding inner center y_{ℓ_i} is modified, while the rest remain all the same. This basic clustering algorithm takes the conventional competitive learning of winner-take-all (WTA) learning [1].

3 Clustering Via Statistical Learning Analysis

In most cases, the performance of the above classical center-based clustering algorithm greatly depends on the number of clusters fixed in advance and contributes to good clustering results only if the number of clusters has already been known as prior knowledge. However, when the number of clusters is unknown beforehand, it will be quite difficult to achieve a reasonable solution.

In order to tackle this problem, an alternative clustering mechanism is directly inspired from statistical learning analysis. The essence of statistical learning analysis is to achieve Structural Risk Minimization instead of Empirical Risk Minimization, as a sound statistical basis for the assessment of model adequacy [5]. Given that the learning model is completely unknown, the goal for clustering here not only concerns the issue of parameter learning, but also attaches great importance to the construction of the predictive models from the data to be learned.

3.1 Membership Hypothesis

One typical membership hypothesis is first specifically considered [7], so that not only the winner would be modified to adapt to the input, but also its rival will receive some penalty:

$$M(y_\ell | x_i) = \begin{cases} 1 & \text{if } \ell = \ell_i \\ -1 & \text{if } \ell = \ell_r \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

$$\ell_i = \arg \min_j \gamma_j d^2(x_i, y_j) \quad ,$$

$$\ell_r = \arg \min_{j, j \neq \ell_i} \gamma_j d^2(x_i, y_j)$$

where $\gamma_j = n_j / \sum_{\ell=1}^K n_\ell$ is the relative winning frequency of the inner representation y_j , as a conscience strategy to reduce the winning probability of certain frequent winners to some extent, and n_j is the cumulative number of the occurrences of $M(y_j | x_t) = 1$ during the past learning. When starting from a number of clusters that is larger than the natural number of groups in the dataset, the aim is to automatically adjust the effective number of clusters.

3.2 Objective Function

On the basis of the above membership hypothesis, the clustering procedure here will not only be determined by the winner, but also by its rival. In other words, for each input, both the corresponding inner centers of the winner and the rival are modified by feed-back, with one for pure learning and the other for penalty.

Heuristically, when taking the membership hypothesis into the objective function of the k-means algorithm, we can update the objective function for optimization as:

$$E = \frac{1}{N} \sum_{t=1}^K \sum_{i=1}^N M(y_\ell | x_t) d^2(x_t, y_\ell) = \frac{1}{N} \sum_{i=1}^N d^2(x_t, y_{\ell_i}) - \frac{1}{N} \sum_{i=1}^N d^2(x_t, y_{\ell_r}) \quad (4)$$

The above function is made up of two parts in the sense of statistical learning, one for empirical risk calculation, the other one for structural assessment. Let the objective function E be decomposed into the plain part E_{MSE} and the additional part E_{SR} , $E = E_{MSE} + E_{SR}$. Minimizing the objective function E will lead to x_t partitioned into the direction of both the minimal similarity with y_{ℓ_i} and maximal dissimilarity with y_{ℓ_r} simultaneously.

In detail, for the benefit of more compatibility in high-dimensional space, here we replace the commonly used Euclidean norm by one higher order metric as the similarity measure so that one of the problems encountered in high-dimensional space, i.e., the ‘‘concentration of measure’’ phenomenon, will be diminished to some degree [8]. And in order to seek a simpler solution as well as to avoid a negative or infinite objective function coming from the structural assessment, we take the Cityblock distance for an easy realization in the second part with only additions, subtractions, and arithmetic comparisons, and turn it into a power fraction expression instead. Inspired by the Minkowski distance metric, the objective function E then becomes

$$E_{MSE} = \frac{1}{Np} \sum_{t=1}^N \left(\sum_{m=1}^d |x_{t,m} - y_{\ell_t,m}|^p \right) \quad (5)$$

$$E_{SR} = \frac{\alpha}{Np} \sum_{t=1}^N \left(\sum_{m=1}^d |x_{t,m} - y_{\ell_r,m}| \right)^{-p}$$

Here d refers as the dimension in the space, $p \geq 2$ defines the order of the average error in the objective function, and α is introduced as a constant factor, $0 < \alpha \leq 1$, to control the influence of the structural assessment. Although E_{SR} is an indispensable part in the model construction of the proposed clustering method, E_{MSE} still plays the most essential role in the whole learning process, which should be attached greater importance to.

3.3 Adaptive Algorithm

With the above selection and modification of mathematical criterion, the derivatives of the objective function E with respect to the center y_ℓ can be computed and an iterative procedure for clustering can then be derived:

$$\frac{\partial E}{\partial y_\ell} = \begin{cases} \frac{\partial E_{MSE}}{\partial y_\ell} = -\frac{1}{N} \sum_{t=1}^N |x_t - y_\ell|^{p-2} (x_t - y_\ell) & \text{if } \ell = \ell_t \\ \frac{\partial E_{SR}}{\partial y_\ell} = \frac{\alpha}{N} \sum_{t=1}^N \text{sgn}(x_t - y_\ell) \left(\sum_{m=1}^d |x_{t,m} - y_{\ell,m}| \right)^{-p-1} & \text{if } \ell = \ell_r \\ 0 & \text{otherwise} \end{cases}, \quad (6)$$

where $\text{sgn}(x_t - y_\ell) = (x_t - y_\ell) / |x_t - y_\ell|$, refers to the sign function that extracts the sign from the difference between the input x_t and the center y_ℓ .

For each input x_t , the adaptive update algorithm for the center y_ℓ is:

$$y_\ell^{new} = y_\ell^{old} + \Delta y_\ell$$

$$\Delta y_\ell = \begin{cases} \eta_t |x_t - y_\ell^{old}|^{p-2} (x_t - y_\ell^{old}) & \text{if } \ell = \ell_t \\ -\eta_r \text{sgn}(x_t - y_\ell^{old}) \left(\sum_{m=1}^d |x_{t,m} - y_{\ell,m}^{old}| \right)^{-p-1} & \text{if } \ell = \ell_r \\ 0 & \text{otherwise} \end{cases}, \quad (7)$$

where η_t and η_r both are constant rates for learning, with $0 < \eta_r \leq \eta_t < 1$. These iterative steps are repeated until one of the two following conditions is fulfilled: either each extra center y_ℓ is pushed far away from the data, or if the clustering results remain roughly fixed for all inputs. When the above rates are appropriately selected, the clustering algorithm has the capacity to not only assign a suitable cluster position

to each input, but also to automatically allocate a proper number of clusters for the input dataset.

4 Simulation Experiment

Simulation experiments on a sample database (Gaussian mixture) were carried out to verify the performance of the proposed clustering algorithm. The experimental dataset consists of a set of samples following a mixture of no more than five Gaussian distributions with different location, mixture proportion and degree of overlap among clusters inside the [-1, 1] domain in a 2-dimensional space. Some examples of datasets are shown in Fig.1.

Given a hypothetical number of clusters larger than the original number of mixtures, both k-means and the proposed algorithm were respectively employed for clustering.

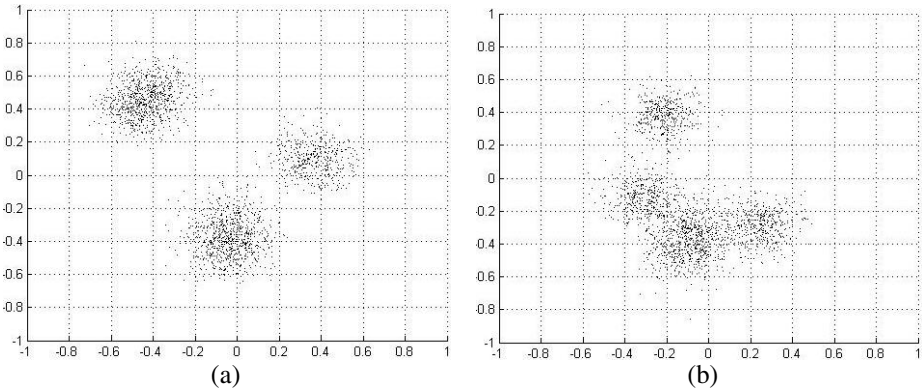


Fig. 1. Dataset examples

5 Result Analysis

Starting from a too large number of clusters (set to eight here), the clustering performances as well as the paths of centers in both k-means and the proposed algorithm for the above example databases are shown as Fig.2. A comparison could be made accordingly between their clustering results. Fig. 2 (a1) and (b1) are the results of the k-means algorithm, and (a2) and (b2) are the results of the clustering algorithm proposed in this paper with the learning rate $\eta_t = 0.005$ and $\eta_t = 0.0005$; (a1) and (a2) refer to dataset (a), and (b1) and (b2) to dataset (b). The clustering algorithm proposed in this paper earned activation for the winners and assigned penalty for their rivals, so that the winners concentrate more around the natural centers of the clusters and their rivals are driven far away from the datasets. Samples from unknown clusters are then assigned to the most competitive clusters, whose centers are representative of the datasets. With adequate parameters, the effective number of clusters can be easily observed, while the extra ones can be identified and removed after or even during

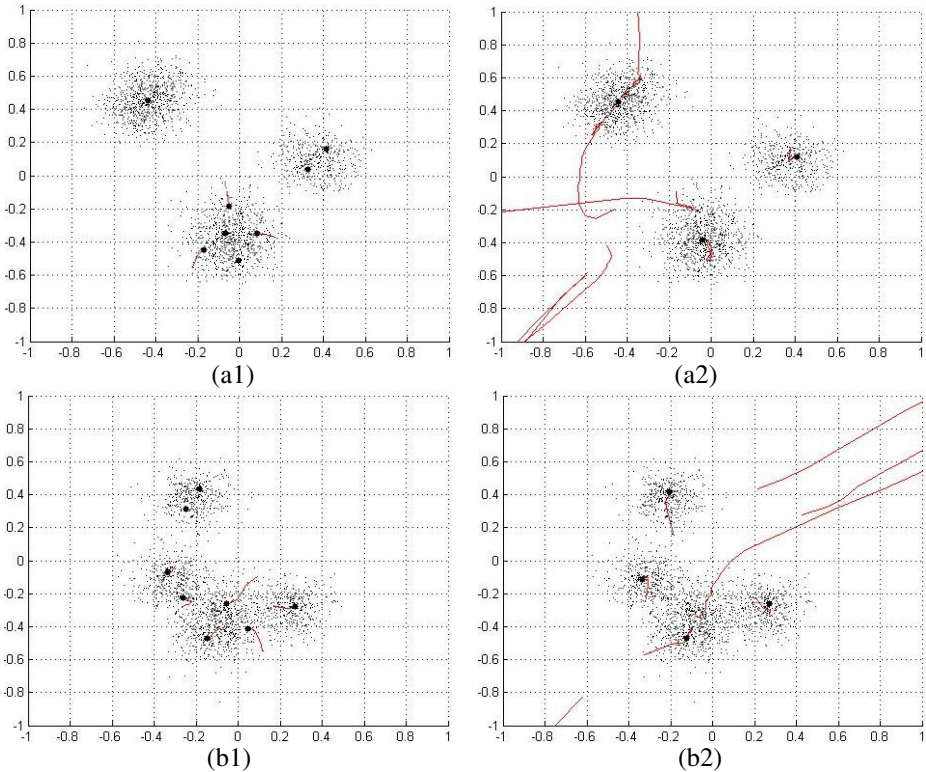


Fig. 2. Clustering performances and paths of cluster centers during learning. The final position is identified by a dot. Fig.2 (a1) and (b1) show the results of the k-means algorithm; (a2) and (b2) show the results of the clustering algorithm proposed in this paper. (a1) and (a2) correspond to dataset (a), and (b1) and (b2) to dataset (b) in Fig.1.

learning. On the contrary, the k-means algorithm maintains the originally given number of clusters, some of them turning out to be meaningless at the end for the correct number was not guessed before learning.

6 Conclusions

In this paper, an alternative for center-based clustering algorithms, in particular the k-means algorithm, is presented via statistical learning analysis. The essence of statistical learning principle, i.e., both the empirical risk and structural assessment, is taken into particular account for the clustering algorithm so as to derive and develop the relevant minimization mathematical criterion with automatic parameter learning and model selection in parallel. The proposed clustering algorithm roughly decides on the number of real clusters, prompts the winner by activation and obstructs its rival by penalty, so that the most competitive center wins for possible prediction and the extra ones are driven far away from the distribution. The only prerequisite is to start with a

number of clusters that exceeds the natural number of clusters in the dataset. Simulation experiments achieve good performances of the double learning tasks in clustering, and show how the number of effective clusters is automatically extracted during learning.

Acknowledgments. Rui Nian is sponsored by the China Scholarship Council for her overseas study in Belgium. This research was fully supported by the Natural Science Foundation of China (60572064).

References

1. Selim, S.Z., Ismail, M.A.: K-Means-Type Algorithms: A Generalized Convergence Theorem and Characterization of Local Optimality. *IEEE Trans. On PAMI-6* 1, 81–87 (1984)
2. Bilmes, J.A.: A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. Technical Report, University of Berkeley, ICSI-TR-97-021 (1997)
3. Zhang, B.: Comparison of the Performance of Center-Based Clustering Algorithms. In: Whang, K.-Y., Jeon, J., Shim, K., Srivastava, J. (eds.) *PAKDD 2003. LNCS (LNAD)*, vol. 2637, pp. 63–74. Springer, Heidelberg (2003)
4. Ma, J.W., Cao, B.: The Mahalanobis Distance Based Rival Penalized Competitive Learning Algorithm. In: Wang, J., Yi, Z., Žurada, J.M., Lu, B.-L., Yin, H. (eds.) *ISNN 2006. LNCS*, vol. 3971, pp. 442–447. Springer, Heidelberg (2006)
5. Vapnik, V.N.: *The Nature of Statistical Learning Theory*. Springer, Berlin (1995)
6. Xu, L.: *Bayesian Ying Yang harmony learning. The handbook of brain theory and Neural Networks*. MIT Press, Cambridge (2002)
7. Xu, L., Krzyzak, A., Oja, E.: Rival Penalized Competitive Learning for Clustering Analysis, RBF Net, and Curve Detection Bayesian Ying Yang harmony learning. *IEEE Trans. Neural Networks* 4, 636–649 (1993)
8. Verleysen, M.: Learning High-dimensional Data. In: Ablameyko, S., et al. (eds.) *Limitations and Future Trends in Neural Computation*, pp. 141–162. IOS Press, Amsterdam (2003)

A Survey on Statistical Pattern Feature Extraction

Shifei Ding^{1,2}, Weikuan Jia³, Chunyang Su¹, Fengxiang Jin⁴, and Zhongzhi Shi²

¹ School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221008

² Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080

³ College of Plant Protection, Shandong Agricultural University, Taian 271018

⁴ College of Geoinformation Science and Engineering, Shandong University of Science and Technology, Qingdao 266510

dingsf@cumt.edu.cn, dingshifei@sina.com

Abstract: The goal of statistical pattern feature extraction (SPFE) is ‘low loss dimension reduction’. As the key link of pattern recognition, dimension reduction has become the research hot spot and difficulty in the fields of pattern recognition, machine learning, data mining and so on. Pattern feature extraction is one of the most challenging research fields and has attracted the attention from many scholars. This paper summarily introduces the basic principle of SPFE, and discusses the latest progress of SPFE from the aspects such as classical statistical theories and their modifications, kernel-based methods, wavelet analysis and its modifications, algorithms integration and so on. At last we discuss the development trend of SPFE.

1 Introduction

With the development of science and technology, the research objects are more and more complex. The complex systems have the characteristics of high dimension and salient nonlinearity. Large amounts of data provide utilizable information, but also make it difficult to use these data effectively. Useful knowledge may be inundated in a large number of redundant data, this will occupy a lot of storage space and computation time, make the training process time-consuming, finally affect the precision of recognition, and cause dimensionality curse. So how to make use of these huge volumes of data, analyze, extract useful information and exclude the influence of related or repeated factors, are the problems that feature extraction needs to solve, that is to reduce the feature dimension under the prerequisite of not affecting the problem solving as much as possible; this provides a good precondition to pattern recognition [1]. Feature extraction is the key link of pattern recognition system; it determines the final results of recognition system.

How to extract efficient and reasonable reduction data from mass datasets while keeping the data completely, that is the connotation of pattern feature extraction. The basic task of feature extraction is to find out a group of the most effective features for classification, so as to design classifier effectively. However in piratical problems it is

often hard to find the most efficient features, this makes feature extraction become one of the most important, difficult and challenging tasks in the fields of pattern recognition system, data mining, machine learning and so on. A large number of domestic and foreign scholars are attracted to this field and they've brought some good results.

Various compression algorithms solve the problems of information feature extraction to some extent, but they have some disadvantages. Many scholars have proposed some new ideas which make the research of pattern feature extraction improve greatly. We'll discuss the research progress of Statistical Pattern Feature Extraction (SPFE).

2 Researches on SPFE

SPFE is to use the existing feature parameters to comprise a lower-dimensional feature space, map useful information contained by original features to a small number of features, ignoring redundant and irrelevant information [2,3]. This is the process of pattern feature extraction, which can be summarized as the process of 'low loss dimension reduction' of original information; mapping is the feature extraction algorithm. Aiming at different problems, we select different feature extraction algorithms which can be roughly divided into linear feature extraction and nonlinear feature extraction. Linear combination is easy to compute, and the early high-dimensional data process methods use linear methods to reduce the dimension. However, in practice we often meet nonlinear, time-varying systems, so the researches on nonlinear feature extraction are more.

2.1 The Methods Based on Statistical Analysis and Their Improved Methods

Statistical analysis theory is the frequently-used method of data feature extraction. It can analyze the statistical laws when several objects and several indices are interrelated; it is a comprehensive analysis method. Statistical methods are based on forceful theory, have lots of algorithms, and can effectively analyze and process the data. Analyzing the data features or classifying the data subsets should subject to statistics irrelevant assumption.

Principal Component Analysis (PCA) is a kind of statistical method that turns various feature indicators to a small number of indicators that describe the data sets from the perspective of the effectiveness of the features. PCA should find several comprehensive factors to replace the original mass variables, here it is requested that the principal components should reflect the information of original data as much as possible, and should be independent from each other, so as to achieve the goal of simplification. PCA represents the principal components as the linear combinations of the single variables, putting emphasis on explaining the total variance of the variables, when the eigenvalue of the given covariance matrix or correlation matrix are only, the principal components are unique in general; however in PCA, the variance can not fully reflect the amount of information. Two-dimensional PCA was proposed based on PCA and was used to extract statistical features of palm prints images [4], it was proved that the generalization ability was better than traditional PCA, based on this,

the paper proposed and defined improved two-dimensional PCA, and proved it could keep the total divergence of the training sample images, and at the same time more effectively extract the sample features. It improved the recognition rate and drastically reduced the feature dimension of original algorithm and the computation complexity, and made the system more practical.

Line Discriminant Analysis (LDA) is a typical representative of linear feature extraction methods; it is widely applied, but is restricted by scared samples problems. A feature extraction algorithm which is based on boundary and is applicable to scared samples problems was proposed [5]. The algorithm used the characteristic that when the sample size of the high-dimensional data was small, the linear separable probability of the data increased and low-dimensional projection of the data tended to be normal distribution, it defined new classificatory borders, not only considered the discrete degrees in-class and between-class brought forward by Linear Discriminant Analysis, and also took the variance difference of each category into account. It obtained optimal projection vector by the maximization of the borders, at the same time avoided scared samples problems caused by in-class discrete degrees singular matrix.

Partial Least Squares (PLS) method emerged and developed in recent years, it is a kind of multi-element data processing method which has wide applicability and is built on the basis of PCA, it is the widening of Ordinary Least Square method. Then information feature compression algorithm [6] based on PLS was advanced, the algorithm can better resolve the difficult problem that observation sample data are few but the explanatory variables are more. If the explanation space "direction" is selected suitably, then the data fitting and forecasting will be robust and reliable. When there is higher level correlation in explaining sets, PLS can use the system data to analyze and sift, extract the integrated variables which can explain the forecasting variables best, and establish appropriate model. Therefore, when the method compresses the explanatory variable data, it takes the related level of forecasting variable into account; its compression results will be more significative.

Projection Pursuit (PP) [7] is used to analyze and process high-dimensional observation data, specially the data from normal population; its basic idea is to project high-dimensional data to one to three dimensional sub space, search the projection that can reflect the structure or features of the high-dimensional data, so as to achieve the goal of studying and analyzing the high-dimensional data. This method is not restricted by the assumption of normal population, and in practice many data do not correspond with normal population; people don't have enough prior knowledge with the data distribution; it overcomes the problems that are brought about by 'the curse of dimensionality', and at the same time the data visibility is increased; it can exclude the interference of that the data structure has no or little relationship with features. M T Gao used genetic algorithm to search the best projection direction, used the projection matrix of optimization projection direction to represent the linear and nonlinear data structure and feature projection of original data [8]. Gao applied this method in text clustering, compared with K-mean clustering and proved the algorithm was effective.

Independent Component Analysis (ICA) [9] is a kind of new statistical method develops in recent years; the goal of the method is that the observation data will be processed with some kinds of linear decomposition, so as to make the data be decomposed to statistical independent components. The basic of ICA is to use a hidden statistical

variable model $x = As$; it represents how the observation data are produced by the mixing of independent components. Independent components are hidden variables, this means they can't be observed directly, and the mixing matrix is supposed to be unknown. What can be observed is only the random vector x , A and s must be estimated, and they must be estimated under the assumed conditions as few as possible.

ICA supposes that components are statistical independent, and the independent components are must supposed to be not Gauss distribution, unbeknown mixed matrix are supposed to be square matrix, if the inverse matrix of A can be ciphered out, to assume as W , then the independent components can be got by $s = Wx$. It can be known that there are two uncertainties of ICA model: we're not sure of the variance of independent components and we can not ensure the order of the independent components.

The independent components analysis algorithm improved by basis function was used in extracting the image features [10]; by analyzing the Laplace priori conditions of the images, the ICA problems are simplified to solving the least normal number; the algorithm doesn't need to optimize the high-order nonlinear compare function and is sparser and has a faster convergence speed. J Karhunen et al used ICA to extract the image pattern feature [11], P C Yuen et al used ICA to do face recognition [12], and these researches show the wide application prospect of ICA. Because ICA appears in recent years, its theories and algorithms are not very mature, many substances should be added and perfected. The rising ICA theories and methods will start another upsurge in the study of pattern feature extraction.

Moreover there're incremental PCA (IPCA) and incremental discriminant analysis, a new incremental face feature extraction method- incremental weighing average samples analysis was used in real time face recognition [13]; the feature extraction method based on singular value decomposition of matrix improved the disadvantages of classical mathematical methods [14].

2.2 Methods Base on Kernel Functions

Kernel thinking [15] is to introduce kernel function to other algorithms, transform the non-linear problem of the original space to a linear problem of feature space, yet the actual calculation run in the original space. Then the method of using kernel function develops a new thinking for solving nonlinear problems, it can be applied to many linear algorithms of data analyzing, especially the algorithms that appear as a form of inner product. The method is based on selecting a conditional function $K(x_i, x_j)$ that is symmetric, continuous and subject to the Mercer theorem, x_i and x_j are the two sample points of the input space, the method is to achieve the mapping $\Phi : R^{d_L} \rightarrow H$ from the input space d_L to d_H -dimension feature space, and there is

$$K(x_i, x_j) = \sum_{n=1}^{d_H} \Phi_n(x_i) \Phi_n(x_j) \quad (1)$$

The aim of achieving mapping is to map the problems difficult to be solved to feature space to process. At present, the kernel function used more are Linear Polynomial Function, p Order Polynomial Function, Gaussian Radial Basis Function (RBF) kernel function, Multi-Layer Perception (MLP) kernel function, and so on.

Such as the Kernel-based Principal Component Analysis (Kernel PCA, KPCA) [16], the main idea is to map from input data x via a nonlinear mapping $\Phi(x)$ to feature space F , and then execute the linear PCA in the feature space F . For the computation of eigenvalue in the feature space and the vector projection in the feature space, KPCA doesn't require the mapping $\Phi(x)$ having explicit format, but only computing the dot product of mapping, actually the dot product can use the kernel function

$$K_{ij} = k(x_i, x_j) = (\Phi(x_i) \cdot \Phi(x_j)) \quad (2)$$

to compute. The nonlinear of KPCA is achieved by kernel transformation, transforming input space to Hilbert feature space, so it can be said that the PCA is computed in the input space, while Kernel PCA in the feature space.

Kernel-Based Fisher Discriminant Analysis[17] (Kernel FDA, KFDA) and Kernel-Based Canonical Correlation Discriminant Analysis [18] (Kernel CCDA, KCCDA), and so on are well referenced kernel functions, they overcome the weaknesses of solving linear problems only, though in form it is a little complicated, but it can turn nonlinear problems into linear, and is easy to resolve the problems. As a bridge from linear to nonlinear, kernel function can generalize the methods that only can solve linear problems to that can solve nonlinear problems.

2.3 Methods Based on the Integration of Several Algorithms

When each method plays its advantages, it also has certain disadvantages, and different methods generally have different adapting environments, it is hard to get good robustness and high precision using only one feature extraction method. Combining various methods organically, developing one's advantage and avoiding one's weakness, feature can be compressed better, better feature information can be provided, and thereby the accuracy of recognition will be improved. R. W. Swiniarski combined PCA and rough set [19], based on reducing the dimension by PCA, he used the attribute reduction algorithms in rough set to compress the dimension further, applied it in the neural network recognition of face images, and achieved ideal results.

PCA feature compressed algorithm based on information theory [20], according to the concept of information function in Shannon's information theory, combining the intrinsic behavior of the eigenvalue, the concept of generalized information function was advanced. And it was applied in feature compression of PCA, the concept of information rate and cumulative information rate was advanced, then the PCA feature compression algorithm that based on information theory was established. The algorithm can describe the level of information compressed better to a large extent; it includes more information content of original features than principal components received by PCA. This algorithm is the combination of the advantages of principal component analysis and information theory.

Fei Zuo et al. proposed cascading the three methods CGD, CTF and CFR and using this method in facial expression recognition [21], the results showed that the performance of the catenation method was far better than the independent method. The two way feature compress method based on PCA and immunity clustering effectively excluded the relativity between various feature parameters [22], the algorithm will perform more effectively and has wider applicability while adding the step of normalization of antigen data, and the step of directly removing the similar samples. Combining iris technique and multi-dimension scale analysis to extract the features will better improve the accuracy of iris recognition [23].

2.4 Other Methods

Pattern feature extraction is always an active study area; many scholars have paid great efforts and made great contributions. Besides the improved methods based on the above classical methods, many scholars have proposed new theories and methods, such as Nonnegative Matrix Factorization (NMF) [24], Locally Linear Embedding (LLE) [25], Manifold Learning (ML) [26] and so on.

NMF method is the research results of nonnegative matrix that was published in the famous magazine《Nature》in 1999. D D Lee and H S Seung proposed a new matrix decomposition idea in this paper, which is Non-negative Matrix Factorization (NMF). NMF is a matrix decomposition method under the constraint conditions that all the elements in the matrix are nonnegative numbers. LLE is a nonlinear dimensionality reduction method; it constructs the reconstruct relationship between each warp beam sample point and its neighboring sample points, keeps the reconstruct relationship unchanging in the process of dimensionality reduction, and retains parts of important features of the high-dimensional measuring space. ML is a type of unsupervised statistics learning problems. A manifold can be simply thought as a topological space, it is locally Euclidean, and its main goal is to find low-dimensional smooth manifold embedded in a high-dimensional observation data space. The research content of ML mainly includes the dimensionality reduction of limited data sets that reserve or highlight special features of the original data; density estimation problems of high-dimensional limited sample points that submitted to a distribution; establishing hidden variable model of high-dimensional observation data that influenced by a small number of potential factors. ML can be divided into methods based on local and global, sometimes it can be divided into spectral methods and non-spectral methods. Although ML is a basic research direction, but for its broad applied prospect, Manifold Learning is increasingly becoming a hot issue in recent years.

3 Prospects

The main significance of feature extraction lies in the “Low Loss Dimensionality Reduction”, which enable the problems tend to be simplified and can be easily computed, or increase computation speed then the learning and training of the system becomes easy. Compared to pattern recognition, it is fundamental and antecedent research. In recent years pattern recognition has been applied to various areas. Pattern feature extraction is the basis of recognizing, and learning, and it plays a key role in

improving the recognition accuracy. Pattern feature extraction is the important part of data mining and machine learning. Aiming at different problems, we use reasonable, reliable, feasible feature extraction methods.

Although the theoretical methods of feature extraction and selection have made a lot of achievements, but some methods are still in theory, failed to be put into practice, the purpose of theory studying is to apply it in practice, in future, one of the research hot spot is how to apply mature theory in practice to deal with practical problems. From the view of application, generally, the systems we meet in practice are nonlinear, time-varying systems, so the current hot research focuses on the research on the theories and algorithms of high-dimensional nonlinear feature extraction.

Acknowledgements. This work is supported by the National Natural Science Foundation of China under Grant no.40574001, the 863 National High-Tech Program under Grant no. 2006AA01Z128, and the Opening Foundation of Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences under Grant no.IIP2006-2.

References

1. Duda, R.O., Hart, P.E.: *Pattern Classification and Scene Analysis*. Wiley, New York (1973)
2. Jenson, D.D., Cohen, P.R.: Multi Plecoem Partitions in Induction Algorithms. *Machine Learning* 3, 309–338 (2000)
3. Foman, G.: An Exnetsive Empirical Study of Feater Selection Metrics for Text Classification. *Journal of Machine Learning Research* 3, 1289–1305 (2003)
4. Li, Q., Qiu, Z.D., Sun, D.M., et al.: Online Palmprint Identification Based on Improved 2D PCA. *Acta Electronica Sinica* 10, 1886–1889 (2005)
5. Huang, R., He, M.Y., Yang, S.J.: A Margin Based Feature Extraction Algorithm for the Small Sample Size Problem. *Chinese Journal of Computers* 7, 1173–1178 (2007)
6. Ding, S.F., Jin, F.X., Shi, Z.Z.: Information Feature Compression Based on Partial Least Squares. *Journal of Computer Aided Design & Computer Graphics* 2, 368–371 (2005)
7. Friedman, J.H., Tukey, J.W.: A Projection Pursuit Algorithm for Exploratory Data Analysis. *IEEE Trans. of Computers* 9, 881–890 (1974)
8. Gao, M.T., Wang, Z.O.: A New Algorithm for Text Clustering Based on Projection Pursuit. In: *Proceedings of the Sixth International Conference on Machine Learning and Cybernetics*, pp. 3401–3405 (2007)
9. Comon, P.: Independent Component Analysis: A New Concept. *Signal Processing* 3, 287–314 (1994)
10. Huang, Q.H., Wang, S., Liu, Z.: Improved Algorithm of Image Feature Extraction Based on Independent Component Analysis. *Opto-Electronic Engineering* 1, 121–125 (2007)
11. Karhunen, J., Hyvarinen, A., Vigario, R., et al.: Applications of Neural Blind Separation to Signal and Image Processing. In: *Proceedings of the IEEE 1997 International Conference on Acoustics, Speech, and Signal Processing*, pp. 131–134 (1997)
12. Yuen, P.C., Lai, J.H.: Face Representation Using Independent Component Analysis. *Pattern Recognition* 3, 545–553 (2001)
13. Song, F.X., Gao, X.M., Liu, S.H.: Dimensionality Reduction in Statistical Pattern Recognition and Low Loss Dimensionality Reduction. *Chinese Journal of Computers* 11, 1915–1922 (2005)

14. Wiener, E., Pedersen, J.O., Weigend, A.S.: A Neural Network Approach to Topic Spotting. In: Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval, pp. 317–332 (1995)
15. Aizerman, M., Branverman, E., Rozonoer, L.: Theoretical Foundations of the Potential Foundation Method in Pattern Recognitions. *Automation and Remote control* 25, 821–837 (1964)
16. Taylor, J.S., Holloway, R., Williams, C.: The Stability of Kernel Principal Components Analysis and Its Relation to the Process Eigenspectrum. In: *Advances in Neural Information Processing Systems*, vol. 15, pp. 383–389 (2003)
17. Mika, S., Retsch, G., et al.: Fisher Discriminant Analysis with Kernels. *Neural Networks for Signal Processing IX*, 41–48 (1999)
18. Sun, P., Xu, Z.B., Shen, J.Z.: Nonlinear Canonical Correlation Analysis for Discrimination Based on Kernel Methods. *Chinese Journal of Computers* 6, 789–795 (2004)
19. Swiniarski, R.W., Skowron, A.: Rough Set Methods in Feature Selection and Recognition. *Pattern Recognition Letters* 6, 833–849 (2003)
20. Ding, S.F., Jin, F.X., Wang, J., et al.: New PCA Feature Compression Algorithm Based on Information Theory. *Mini-micro Systems* 4, 694–697 (2004)
21. Zuo, F., De With, P.H.N.: Facial Feature Extraction Using a Cascade of Model-Based Algorithms. In: *Proceedings IEEE Conference on Advanced Video and Signal Based Surveillance*, pp. 348–353 (2005)
22. Fan, Y.P., Chen, Y.P., Sun, W.S., et al.: Algorithm for Bi-directional Reduce Feature Data Based on the Principal Component Analysis and Immune Clustering. *Acta Simulata Systematica Sinica* 1, 148–153 (2005)
23. Makram, N., Bouridane, A.: An Effective and Fast Iris Recognition System Based on a Combined Multiscale Feature Extraction Technique. *Pattern Recognition* 3, 868–879 (2008)
24. Lee, D.D., Seung, H.S.: Learning the Parts of Objects by Non-negative Matrix Factorization. *Nature* 6755, 788–791 (1999)
25. Roweis, S.T., Saul, L.K.: Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science* 5500, 2323–2326 (2000)
26. Seung, H.S., Lee, D.D.: The Manifold Ways of Perception. *Science* 12, 2268–2269 (2000)

Nonlinear Innovation to Noisy Blind Source Separation Based on Gaussian Moments

Hongjuan Zhang², Chonghui Guo¹, Zhenwei Shi³, and Enmin Feng²

¹ Institute of Systems Engineering, Dalian University of Technology,
Dalian 116024, P.R. China
guochonghui@tsinghua.org.cn

² Department of Applied Mathematics, Dalian University of Technology,
Dalian 116024, P.R. China
zhhj2108@yahoo.com.cn,
emfeng@dlut.edu.cn

³ Image Processing Center, School of Astronautics, Beijing University of Aeronautics
and Astronautics, Beijing 100084, P.R. China
shizhenwei@mail.tsinghua.edu.cn

Abstract. This paper addresses blind source separation problem for noisy data based on the concepts of nonlinear innovation and Gaussian moments. An objective function which incorporates Gaussian moments and the nonlinear innovation of original sources is developed. Minimizing this objective function, a noisy blind source separation algorithm is proposed when the noise covariance is known and source signals are non-stationary in the sense that the variance of each is assumed to change smoothly as a function of time. In addition, this method is further extended to the case of noise covariance unknown. Validity and performance of the described approaches are demonstrated by computer simulations.

1 Introduction

Over the past decades the problem of blind source separation(BSS) [1,2,3,4] has received much research attentions because of its potential applicability to a wide range of problems, such as communications signals and biomedical signals analysis and processing, and so on. Many BSS methods using the statistical properties of primary sources have been proposed, such as non-Gaussianity [2,5,6,7], linear predictability or smoothness [1,8], code complexity [9,10,11] and nonstationarity of variance [12]. It must be pointed that most of the previous researches have been conducted with the assumption of no additive noise or that noise can be neglected. However, noise is an unavoidable factor in real world applications. Therefore, it needs to be taken into account when designing algorithms. And many literatures have considered a special noise case for estimating the independent component analysis(ICA) data model when the independent components(ICs) are time-dependent source signals [11,13]. In addition, some related works [4,14], which considered the separation of the noisy signals without time

structures, were presented. These techniques are based on the measures of the non-Gaussianity, such as entropy-based measures or cumulant-based measure like kurtosis of the signals. Recently, Shi et al. [15] presented a simple algorithm (simplified with "MINDIFF" algorithm), which aims to exploit the nonstationarity of sources in the sense that the variance of each independent source signal is assumed to change smoothly as a function of time. This method is based on a new simple principle: minimizing the loss function of the nonlinear innovation of original sources, which is a special case of "generalized complexity pursuit". The efficient implementation of this method has been demonstrated by simulations, especially its robustness to the outliers [15]. However, MINDIFF method has been designed for the case of noise-free mixtures and does not consider the case when noise exists. Hence, it is necessary to improve it to the noise case.

In this paper, we introduce a novel noisy BSS approach based on nonlinear innovation and Gaussian moments. An objective function is proposed, which incorporates Gaussian moments and the nonlinear innovation of original sources. Minimizing this objective function, a simple blind source separation algorithm is presented. In this method, the effect of noise can be removed directly from the cost function. It can be considered an extension of MINDIFF algorithm when Gaussian noise is present and independent components are nonstationary in the sense that the variance of each independent source signal is assumed to change smoothly as a function of time. Experiments in the following show that the proposed algorithm outperforms some existing algorithms, for example, noise techniques—the algorithm introduced by Hyvärinen [14] (called "FastNoisy-ICA"), the algorithm proposed by Shi et al. [11] (simplified with "NoisyCP"), the algorithm in [13] (called "NoisyUN" or "New NoisyUN") and noise-free approach—MINDIFF algorithm [15].

2 Proposed Algorithm

2.1 Quasi-whitening

Denote the observed sensor signals $\mathbf{x}(t) = (x_1(t), \dots, x_n(t))^T \in \mathcal{R}^n$ by equation

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t), \quad (1)$$

where \mathbf{A} is an $n \times n$ unknown mixing matrix, $\mathbf{s}(t) = (s_1(t), \dots, s_n(t))^T \in \mathcal{R}^n$ is a vector of unknown temporally correlated sources (zero-mean and unit-variance) and $\mathbf{n}(t) = (n_1(t), \dots, n_n(t))^T$ is a vector of additive Gaussian noise which is modeled as a stationary, zero-mean random process independent of the sources, and whose covariance matrix is defined as $\mathbf{\Sigma} = E\{\mathbf{n}(t)\mathbf{n}(t)^T\}$.

It must be noted that, in the preliminary whitening, the effect of noise must be considered. This is quite simple if the noise covariance matrix $\mathbf{\Sigma}$ is known or can be estimated. Provided that the measured sensor signals $\mathbf{x}(t)$ has the covariance matrix $\mathbf{C} = E\{\mathbf{x}(t)\mathbf{x}(t)^T\}$, the ordinary whitening should be changed into the following "quasi-whitening" operation

$$\tilde{\mathbf{x}}(t) = (\mathbf{C} - \mathbf{\Sigma})^{-\frac{1}{2}}\mathbf{x}(t). \quad (2)$$

The quasi-whiten data $\tilde{\mathbf{x}}$ follows a noisy ICA model as well, that is $\tilde{\mathbf{x}}(t) = \mathbf{B}\mathbf{s}(t) + \tilde{\mathbf{n}}(t)$, where $\mathbf{B} = (\mathbf{C} - \Sigma)^{-\frac{1}{2}}\mathbf{A}$ is an orthogonal mixing matrix, and noise $\tilde{\mathbf{n}}$ is a linear transform of the original noise which has the following covariance $\tilde{\Sigma} = E\{\tilde{\mathbf{n}}(t)\tilde{\mathbf{n}}(t)^T\} = (\mathbf{C} - \Sigma)^{-\frac{1}{2}}\Sigma(\mathbf{C} - \Sigma)^{-\frac{1}{2}}$ [14].

2.2 Contrast Function

Denote the noise-free data by $\mathbf{y}(t) = \mathbf{B}\mathbf{s}(t)$, we consider predictive coding of a scalar signal $z(t) = \mathbf{w}^T\mathbf{y}(t)$. Suppose the value of $z(t)$ is predicted from the preceding values by some function to be specified

$$h(z(t)) = \sum_{\tau=1}^L \alpha_{\tau}h(z(t - \tau)) + \delta(t), \tag{3}$$

where τ is a specific time delay, α_{τ} is a predicting parameter, L is the degree of the predicting model and h is a differentiable function which defines the predictor structure of the signal. To code the actual value $z(t)$, the innovation

$$\delta(t) = h(z(t)) - \sum_{\tau=1}^L \alpha_{\tau}h(z(t - \tau)), \tag{4}$$

is coded by a scalar quantization method. In fact, assume that we choose only one predicting term and the predicting parameter α_{τ} is 1. The minimization of loss function about the linear or nonlinear innovation of the desired signal

$$\min_{\|\mathbf{w}\|=1} \Psi(\mathbf{w}) = E\{G(h(z(t)) - h(z(t - \tau)))\}, \tag{5}$$

is able to obtain, where functions h and G are chosen just like literatures [15].

In MINDIFF method, however, the noise term is not considered. Therefore, we might extend this method to noise case. Fortunately, it could be used for noisy data if only we are able to estimate $\Psi(\mathbf{w})$ of the noise-free data from noisy observation \mathbf{x} . The main point of this problem is to select some suitable measures which are immune to Gaussian noise. Denote by

$$\varphi_c(x) = \frac{1}{c}\varphi\left(\frac{x}{c}\right) = \frac{1}{\sqrt{2\pi}c} \exp\left(-\frac{x^2}{2c^2}\right) \tag{6}$$

the Gaussian density function of variance c^2 , and by $\varphi_c^{(k)}(x)$ the k -th ($k > 0$) derivative of $\varphi_c(x)$. Denote further by $\varphi_c^{(-k)}(x)$ the k -th integral function of $\varphi_c(x)$, obtained by $\varphi_c^{(-k)}(x) = \int_0^x \varphi_c^{(-k+1)}(\xi)d\xi$, where we define $\varphi_c^{(0)}(x) = \varphi_c(x)$. Then we have the following theorem [11][14]

Theorem 1. *Let v be any non-Gaussian random variable, and denote by n an independent Gaussian noise variable of variance σ^2 . Define the Gaussian function φ as in (6). Then for any constant $c > \sigma^2$, we have*

$$E\{\varphi_c(v)\} = E\{\varphi_d(v + n)\}, \tag{7}$$

with $d = \sqrt{c^2 - \sigma^2}$. Moreover, (7) still holds when φ is replaced $\varphi^{(k)}$ for any integer index k .

This theorem means that we can estimate the noisy model by minimizing, for the quasi-whitened data $\tilde{\mathbf{x}}$, the following function

$$\begin{cases} \min & \Phi(\mathbf{w}^T \tilde{\mathbf{x}}(t)) = E\{G(\varphi_{d(\mathbf{w})}^{(k)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) - \varphi_{d(\mathbf{w})}^{(k)}(\mathbf{w}^T \tilde{\mathbf{x}}(t - \tau)))\}, \\ \text{s.t.} & \mathbf{w}^T \mathbf{w} = 1. \end{cases} \quad (8)$$

where $d(\mathbf{w}) = \sqrt{c^2 - \mathbf{w}^T \tilde{\Sigma} \mathbf{w}}$.

2.3 Learning Algorithm

Before we derive the learning algorithm, first note that

$$\varphi_c^{(k)}(x) = \varphi^{(k)}\left(\frac{x}{c}\right)c^{(-k-1)}, \quad (9)$$

and denote

$$d(\mathbf{w}) = \sqrt{c^2 - \mathbf{w}^T \tilde{\Sigma} \mathbf{w}}, \quad (10)$$

$$X = \varphi_{d(\mathbf{w})}^{(k)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) - \varphi_{d(\mathbf{w})}^{(k)}(\mathbf{w}^T \tilde{\mathbf{x}}(t - \tau)), \quad (11)$$

$$\tilde{d} = c^2 - \mathbf{w}^T \tilde{\Sigma} \mathbf{w}. \quad (12)$$

Then the gradient of term $G(X)$ with respect to \mathbf{w} can be computed as

$$\nabla_{\mathbf{w}} G(X) = g(X)(\nabla_{\mathbf{w}} \varphi_{d(\mathbf{w})}^{(k)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) - \nabla_{\mathbf{w}} \varphi_{d(\mathbf{w})}^{(k)}(\mathbf{w}^T \tilde{\mathbf{x}}(t - \tau))), \quad (13)$$

where

$$\begin{aligned} \nabla_{\mathbf{w}} \varphi_{d(\mathbf{w})}^{(k)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) &= \varphi_{d(\mathbf{w})}^{(k+1)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) \tilde{\mathbf{x}}(t) + \frac{1}{2} d^{-2}(\mathbf{w}) \varphi_{d(\mathbf{w})}^{(k+1)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) \\ &\quad \times \mathbf{w}^T \tilde{\mathbf{x}}(t) \nabla_{\mathbf{w}} \tilde{d} + \frac{(k+1)}{2} d^{-2}(\mathbf{w}) \varphi_{d(\mathbf{w})}^{(k)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) \nabla_{\mathbf{w}} \tilde{d}. \end{aligned} \quad (14)$$

To proceed, we use the lemma [14] to imply

$$d^{-2}(\mathbf{w}) X \varphi_{d(\mathbf{w})}^{(k+1)}(X) + d^{-2}(\mathbf{w}) (k+1) \varphi_{d(\mathbf{w})}^{(k)}(X) = -\varphi_{d(\mathbf{w})}^{(k+2)}(X). \quad (15)$$

This means the gradient in (14) can be expressed as

$$\nabla_{\mathbf{w}} \varphi_{d(\mathbf{w})}^{(k)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) = \varphi_{d(\mathbf{w})}^{(k+1)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) \tilde{\mathbf{x}}(t) - \frac{1}{2} \varphi_{d(\mathbf{w})}^{(k+2)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) \nabla_{\mathbf{w}} \tilde{d}. \quad (16)$$

That is

$$\nabla_{\mathbf{w}} \varphi_{d(\mathbf{w})}^{(k)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) = \varphi_{d(\mathbf{w})}^{(k+1)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) \tilde{\mathbf{x}}(t) - \varphi_{d(\mathbf{w})}^{(k+2)}(\mathbf{w}^T \tilde{\mathbf{x}}(t)) \tilde{\Sigma} \mathbf{w}. \quad (17)$$

Similarly, we can obtain the gradient of $\varphi_{d(\mathbf{w})}^{(k)}(\mathbf{w}^T \tilde{\mathbf{x}}(t - \tau))$ with respect to \mathbf{w} as follows

$$\nabla_{\mathbf{w}} \varphi_{d(\mathbf{w})}^{(k)}(\mathbf{w}^T \tilde{\mathbf{x}}(t - \tau)) = \varphi_{d(\mathbf{w})}^{(k+1)}(\mathbf{w}^T \tilde{\mathbf{x}}(t - \tau)) \tilde{\mathbf{x}}(t - \tau) - \varphi_{d(\mathbf{w})}^{(k+2)}(\mathbf{w}^T \tilde{\mathbf{x}}(t - \tau)) \tilde{\Sigma} \mathbf{w}. \tag{18}$$

Note that we could adapt parameter c before every step so that $d(\mathbf{w}) = \sqrt{c^2 - \mathbf{w}^T \tilde{\Sigma} \mathbf{w}} = 1$ [14] and choose $F(u) = \varphi^{(k)}(u)$ correspondingly. Thus we obtain the finally form of the learning rule

$$\mathbf{w} \leftarrow \mathbf{w} - \mu_{\mathbf{w}} E \{ g(F(\mathbf{w}^T \tilde{\mathbf{x}}(t)) - F(\mathbf{w}^T \tilde{\mathbf{x}}(t - \tau))) (f(\mathbf{w}^T \tilde{\mathbf{x}}(t)) \tilde{\mathbf{x}}(t) - f(\mathbf{w}^T \tilde{\mathbf{x}}(t - \tau)) \tilde{\mathbf{x}}(t - \tau) - f'(\mathbf{w}^T \tilde{\mathbf{x}}(t)) \tilde{\Sigma} \mathbf{w} + f'(\mathbf{w}^T \tilde{\mathbf{x}}(t - \tau)) \tilde{\Sigma} \mathbf{w}) \}, \tag{19}$$

where functions f and g are the derivative of F and G , f' and g' correspond to the derivative of f and g respectively. G is the same as that of MINDIFF algorithm. Example of choice is $F(u) = \log \cosh(u)$, which is an approximation of $\varphi^{(-2)}$ and has been widely used in ICA [12][13][14].

In fact, the assumption that the noise covariance matrix is known in advance would be restrictive. Therefore, we develop our method to the case of noise covariance matrix unknown. Similar to (19), we obtain the iteration form of Σ

$$\Sigma \leftarrow \Sigma - \mu_{\Sigma} \frac{1}{2} E \{ g(F(\mathbf{w}^T \tilde{\mathbf{x}}(t)) - F(\mathbf{w}^T \tilde{\mathbf{x}}(t - \tau))) [(f'(\mathbf{w}^T \tilde{\mathbf{x}}(t - \tau)) - f'(\mathbf{w}^T \tilde{\mathbf{x}}(t))) \mathbf{w}^T (\mathbf{C} - \Sigma)^{-1} (\mathbf{I} + \tilde{\Sigma} \mathbf{w})] \}. \tag{20}$$

Firstly, initialize $\Sigma = \mathbf{0}$ for the procedure of quasi-whitening and then for the estimation of \mathbf{w} according to (19); Based on this new estimation value of \mathbf{w} , update Σ according to (20) and apply it for the next quasi-whitening and the estimation of \mathbf{w}, \dots , and all that. Note that the algorithms given in (19) and (20) are suitable for the noisy data when the energy of noise is not known in advance, which is more reasonable to many practical applications. In the remainder of this paper, the proposed algorithm (19) is denoted as "NoisyMINDIFF" algorithm, and the algorithms (19) and (20) are simplified with "New-NoisyMINDIFF" algorithm together.

3 Simulations

Extensive computer simulations were carried out to verify the validity of the proposed algorithm. In the comparisons, the quasi-whitening data were applied as the inputs of all the algorithms and the unfiltered version of the extracted signals as the outputs of all noisy algorithms. Moreover, the performance of all algorithms is measured by performance index (PI), which is defined [5]

$$PI = \frac{1}{n^2} \left\{ \sum_{i=1}^n \left(\sum_{j=1}^n \frac{|p_{ij}|}{\max_k |p_{ik}|} - 1 \right) + \sum_{j=1}^n \left(\sum_{i=1}^n \frac{|p_{ij}|}{\max_k |p_{kj}|} - 1 \right) \right\}, \tag{21}$$

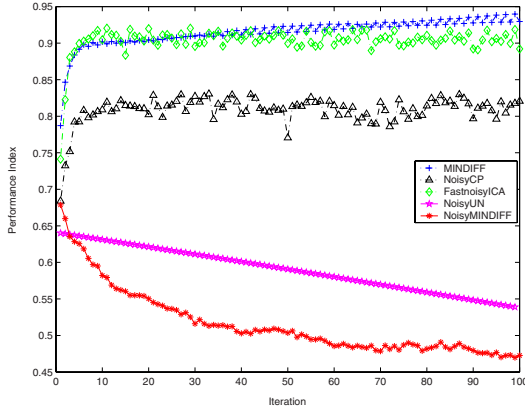


Fig. 1. Comparisons of average PIs to five algorithms when the noise power is 0.25

where p_{ij} denotes the ij th element of $n \times n$ matrix $\mathbf{P} = \mathbf{W}^T \mathbf{B} \mathbf{A}$, where \mathbf{W} is the demixing matrix. PI is zero when the desired signal is perfectly extracted.

In the following experiments, we considered the separation of five artificial signals which had smoothly changing variances as follows (with Gaussian marginal distributions, zero linear autocorrelations and square temporal autocorrelations), which have been applied in some literatures [12,15]. The five signals using a first-order autoregressive model with constant variances of the innovations [9,10], with 5000 time points and Gaussian innovations, which had constant unit variance. The signals had identical autoregressive coefficients (0.8) and the signs of them were completely randomized by multiplying each signal by a binary i.i.d. signal that took the values ± 1 with equal probabilities. The observations \mathbf{x} with a white Gaussian noise were generated by a 5×5 random mixing matrix. We first compared the NoisyMINDIFF algorithm with three noise techniques– FastNoisyICA, NoisyCP, NoisyUN algorithm and one noise free method– MINDIFF algorithm when the noise level is 0.25. The performance was estimated as the average PI values of 50 independent trials. At every trial, five algorithms were run with 100 iterations respectively, which seemed to be always enough for convergence. Here \mathbf{A} and \mathbf{W} were initialized randomly. In the NoisyMINDIFF algorithm, We adopted $G(\cdot) = F(\cdot) = \log \cosh(\cdot)$, the learning rate $\mu_{\mathbf{w}} = 0.01$. All algorithms except FastnoisyICA chose the time delay as $\tau = 1$. Note that the step sizes of NoisyCP algorithm and MINDIFF algorithm were all taken equal to 1 and 0.1 was chosen in NoisyUN algorithm. The nonlinear functions of four existing algorithms were chosen as $\log \cosh(\cdot)$. Fig.1 provides the average performance indexes over 50 independent trials versus iteration numbers. Obviously, NoisyMINDIFF algorithm has the lowest PI values and the best convergence than other algorithms. It must be noted that the performance of FastNoisyICA and NoisyCP algorithm are poor, which can be attributed to the nonstationarity of sources used here and further explained these two algorithms are not capable of exploiting the nonstationarity of sources.

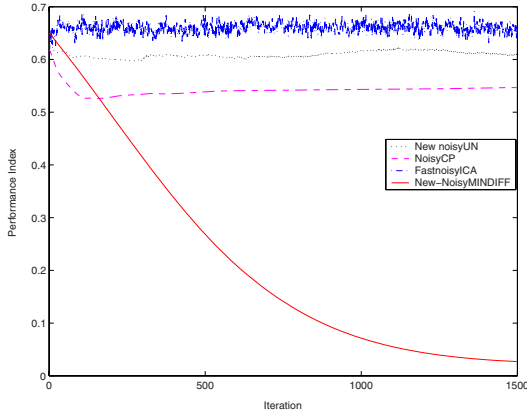


Fig. 2. Comparisons of average PIs to four noisy algorithms when the noise power is unknown

In addition, we investigated the robustness of the proposed algorithm. Twenty outliers whose values were 10 were randomly added in each source signal. We gave here the mean values of 50 independent trials. The mean PI values were 0.8577 (MINDIFF), 0.8479 (FastNoisyICA), 0.7808 (NoisyCP), 0.5741 (NoisyUN) and 0.4927 (NoisyMINDIFF) respectively. Therefore, the NoisyMINDIFF algorithm outperformed other algorithms even if the outliers were introduced.

Fig.2 gives the averaged PIs over 50 independent trials when the noise covariance matrix is unknown in advance. In these simulations, firstly initializing Σ as $\mathbf{0}$ (in fact, observation signals including white Gaussian noises with power 0.25), then the proposed New-NoisyMINDIFF and three existing noisy algorithms were used for the source separation. Note that New NoisyUN was the extension of NoisyUN algorithm to noise unknown. In New-NoisyMINDIFF algorithm, $G(u) = F(u) = \log \cosh(u)$, $\mu_w = 0.02$, $\mu_\Sigma = 0.0005$ and $\tau = 1$. The step size of NoisyCP algorithm was taken equal to 1 and $\mu_w = 0.05$, $\mu_\Sigma = 0.002$ were chosen in New NoisyUN algorithm. The nonlinear functions of NoisyCP, FastNoisyICA and New NoisyUN algorithm were chosen as $\log \cosh(\cdot)$. Fig.2 shows that the proposed algorithm is superior to three existing noisy algorithms at all times, which depicts its satisfying performance even though the noise power is unknown in advance. Moreover, from the results of Fig.1 and Fig.2, we also compared the mean PIs of NoisyMINDIFF algorithm and New-NoisyMINDIFF algorithm, which were 0.51581 and 0.2161 respectively. It interestingly shows that the New-NoisyMINDIFF algorithm performs better than the other, which further indicates the validity of the New-NoisyMINDIFF algorithm.

4 Discussions and Conclusions

A novel approach for the problem of noisy data separation is introduced in this paper, which can be considered as an extension of MINDIFF algorithm

when Gaussian noise is present and independent components are nonstationary in the sense that the variance of each independent source signal is assumed to change smoothly as a function of time. Based on the nonlinear innovations of original sources and the useful property of Gaussian moments, we develop an objective function, which combining Gaussian moments into the nonlinear innovation expression is proposed. Minimizing this objective function, we firstly derive the NoisyMINDIFF algorithm. Furthermore, we also provide the learning rule of the noise level when the noise covariance matrix is unknown, and then give the New-NoisyMINDIFF algorithm. Simulations on artificial signals have shown the superiority of the proposed algorithms.

Acknowledgments. This work is supported by Natural Science Foundation of China under grant No. 10571018, No. 60605002, No. 70431001.

References

1. Cichocki, A., Amari, S.: Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications. Wiley, New York (2002)
2. Comon, P.: Independent Component Analysis: A New Concept? *Signal Process* 36, 287–314 (1994)
3. Cruces-Alvarez, S.A., Cichocki, A., Amari, S.: From Blind Signal Extraction to Blind Instantaneous Signal Separation: Criteria, Algorithm, and Stability. *IEEE Trans. Neural Networks* 15(4), 859–873 (2004)
4. Hyvärinen, A., Karhunen, J., Oja, E.: Independent Component Analysis. Wiley, New York (2001)
5. Amari, S., Cichocki, A., Yang, H.: A New Learning Algorithm for Blind Source Separation. In: *Advances in Neural Information Processing System*, vol. 8, pp. 757–763. MIT Press, Cambridge (1996)
6. Hyvärinen, A.: Fast and Robust Fixed-point Algorithms for Independent Component Analysis. *IEEE Trans. Neural Networks* 10(3), 626–634 (1999)
7. Jutten, C., Herault, J.: Blind Separation of Sources, part I: An Adaptive Algorithm Based on Neuromimetic Architecture. *Signal Process* 24, 1–10 (1991)
8. Barros, A.K., Cichocki, A.: Extraction of Specific Signals with Temporal Structure. *Neural Comput.* 13(9), 1995–2003 (2001)
9. Hyvärinen, A.: Complexity Pursuit: Separating Interesting Components from Time Series. *Neural Comput.* 13, 883–898 (2001)
10. Shi, Z., Tang, H., Tang, Y.: A Fast Fixed-point Algorithm for Complexity Pursuit. *Neurocomput.* 64, 529–536 (2005)
11. Shi, Z., Zhang, C.: Gaussian Moments for Noisy Complexity Pursuit. *Neurocomput.* 69, 917–921 (2006)
12. Hyvärinen, A.: Blind Source Separation by Nonstationarity of Variance: A Cumulant-based Approach. *IEEE Trans. Neural Networks* 12(6), 1471–1474 (2001)
13. Yang, Y., Guo, C.: Gaussian Moments for Noisy Unifying Model. *Neurocomput* (2008) doi:10.1016/j.neucom.2008.03.005
14. Hyvärinen, A.: Gaussian Moments for Noisy Independent Component Analysis. *IEEE Signal Process. Lett.* 6(6), 145–147 (1999)
15. Shi, Z., Zhang, C.: Nonlinear Innovation to Blind Source Separation. *Neurocomput.* 71, 406–410 (2007)

Removal of Artifacts in Electroencephalogram Using Adaptive Infomax Algorithm of Blind Source Separation

Wanyou Guo¹, Liyu Huang¹, Li Gao¹, Tianqiao Zhu¹, and Yuangui Huang²

¹ Department of Biomedical Engineering, Xidian University, Xi'an, 710071, China
huangly@mail.xidian.edu.cn

² Xijing Hospital, The Fourth Military Medical University, Xi'an, 710032, China

Abstract. Infomax algorithm is one of the main strategies in blind source separation. The principle and improvement of the algorithm are introduced firstly in this paper. Nineteen-channel Electroencephalograms (EEGs) which include electromyogram, eye-movement and some other artifacts were decomposed by using this algorithm. Afterwards, three kinds of nonlinear parameters were calculated for all the independent components, and artifact components can be identified automatically by threshold settings. Finally, putting all the artifact components into zero, and projecting the other components to the scalp electrodes, then the purer Electroencephalograms can be gained. The study shows that the various artifacts can be separated from the EEGs successfully with the use of adaptive Infomax algorithm and removal of artifacts can be realized by signal reconstruction. Adaptive Infomax algorithm is a potential tool in removal of artifacts in physiological signal.

1 Introduction

Blind source separation (BSS) is a signal processing technique that has demonstrated the ability of separating sources from mixed recorded signals^[1]. Infomax algorithm is one of the main strategies in blind source separation. Its essential principle is based on the optimization criterion of maximizing the information. Specifically, it means adjusting the un-mixing matrix by the optimization algorithm in order to maximize the output entropy of a neural network. The precondition in processing is that the source signals are all non-Gaussian signals which are statistical independent with one another. The ultimate intention is to separate the independent sources from the mixed signals^[1, 2].

In the processing of biomedical signals, a great deal of physiological signals that have been measured by the researchers are always linear-weighted iterative of some independent components. For example, in the recording of electroencephalogram (EEG), the signals, which are recorded by the electrodes on the scalp, not only contain the electrical activity of the nerve cells in the brain, but also contain some artifacts. The artifacts include the high-frequency artifacts, eye blink artifacts, eyeball movement, electrocardiogram (ECG), electromyography (EMG) and so on. These artifacts

are overlapped in the recorded EEG signals, so it is necessary to remove these artifacts in order to make a more accurate estimation from purer EEG signals. Generally speaking, it is significant to separate the components of physiological meanings from the recorded physiological signals by removing the artifacts. It is thought that the signals and artifacts are produced by mutual independent sources, thus it is a reasonable choice to achieve the artifacts removal of biomedical signals by blind source separation^[3, 4].

The rest of the Paper is organized as follows: In Section 2, the adaptive infomax algorithm for BSS is proposed. Then in section 3 this algorithm is used to analyze the recorded EEG data, and the artifacts are distinguished automatically according to the thresholds that have been initialized by the analysis of three nonlinear parameters. The discussion in Section 4 covers related work and some conclusions.

2 Materials and Data Acquisition

The experiment data derive from the RIKEN brain science institute in Japan^[5]. The scalp electrodes are allocated based on the 10-20 channel system of the international standard. In this experiment nineteen-channel EEG data have been acquired, its sampling frequency is 100Hz and the sampling time is 20s. The locations of the scalp electrodes are respectively FP1、FP2、F7、F3、Fz、F4、F8、T7、C3、Cz、C4、T8、P7、P3、Pz、P4、P8、O1、O2. The original nineteen-channel EEG signals which have not been processed are shown in figure 1. In figure 1, it can be found out that the data gained from channels of FP1 and FP2 contain obvious artifacts of eyeball movements. Further more, the signals that recorded on channels of F3 and Cz are overlapped by some high-frequency artifacts, and the C3-channel is disturbed seriously by muscle activities.

3 Method

The basic idea of adaptive infomax algorithm is to introduce a nonlinear function $r_i = g_i(y_i)$ at the end of each output component y_i but not to estimate its high-order statistic. The criterion is to maximize the total entropy $H(\vec{r})$ of the output $\vec{r} = [r_1, r_2, \dots, r_M]$ after the presentation of a suitable $g_i(y_i)$. $p(y_i)$ is the probability density function of the output component y_i . $p(y_i)$ must be close to $p(s_i)$ in order to make the output \vec{y} approach to the source \vec{s} . That is, $g_i(y_i)$ must approximate to the cumulative distribution function of the source signals^[6, 7]. However, $p(s_i)$, which expresses the probability density function of the source signal s_i , is always unknown. In this paper, an optimization infomax algorithm that has the efficiency of adaptive control is introduced. The principle is illustrated in figure 2.

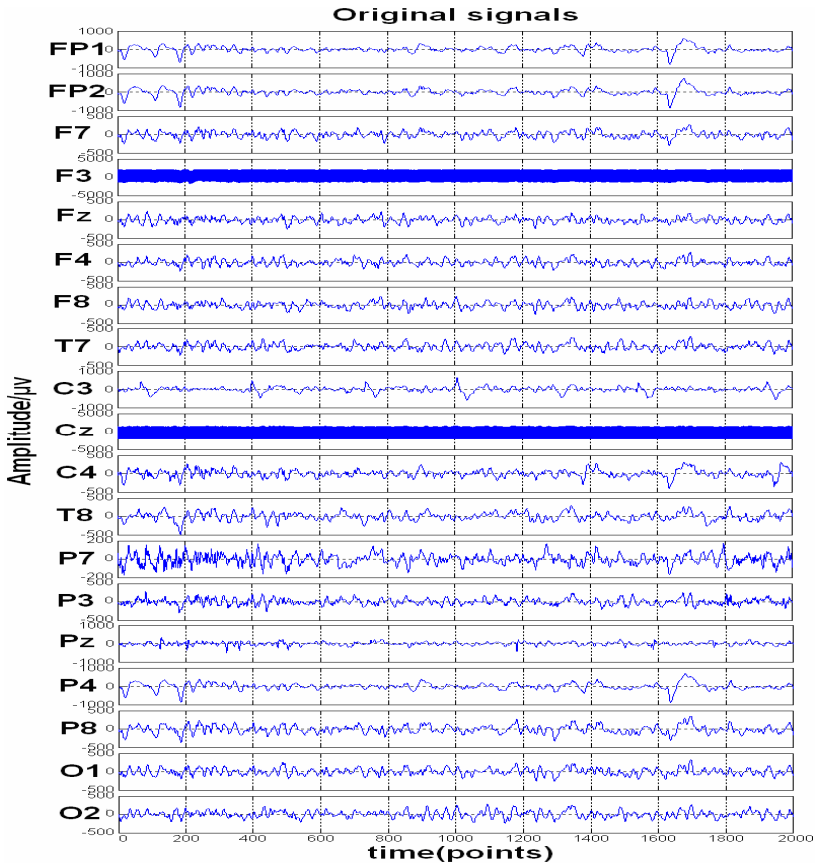


Fig. 1. Original EEG data containing various artifacts

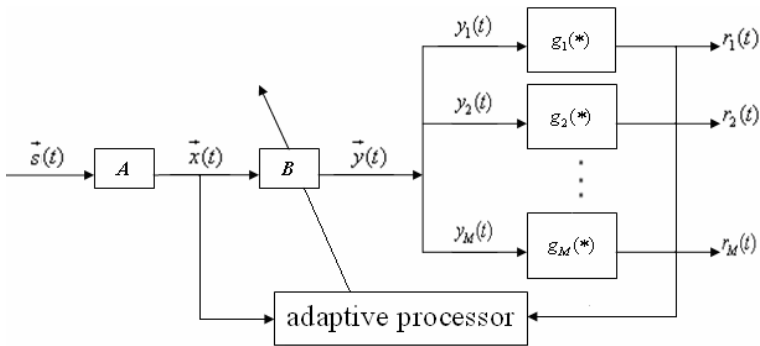


Fig. 2. Principle frame of adaptive infomax algorithm

In figure 2, $\vec{s}(t)$ are the original independent sources, \mathbf{A} is a mixing matrix, $\vec{x}(t) = \mathbf{A} \cdot \vec{s}(t)$ are mixed input signals, \mathbf{B} is an un-mixing matrix, and $\vec{y}(t) = \mathbf{B} \cdot \vec{x}(t)$ are independent output signals. The adjustment of parameters in this optimization algorithm is directed by the following formula:

$$\Delta \mathbf{B} = \mu \frac{\partial H(\vec{r}, \mathbf{B})}{\partial \mathbf{B}} \tag{1}$$

The calculation of the mean can be canceled when the signals are processed by random gradient algorithm. Accordingly, an equation can be gained as follows:

$$H(\vec{r}, \mathbf{B}) = H(\vec{x}) + \log |\mathbf{B}| + \sum_{i=1}^M \log g'_i(y_i) \tag{2}$$

In addition,

$$\frac{\partial H(\vec{r}, \mathbf{B})}{\partial \mathbf{B}} = \mathbf{B}^{-T} - \psi(\vec{y}) \vec{x}^T \tag{3}$$

where

$$\psi(\vec{y}) = \left[-\frac{g''_1(y_1)}{g'_1(y_1)}, \dots, -\frac{g''_M(y_M)}{g'_M(y_M)} \right] \tag{4}$$

So the regulating formula is:

$$\mathbf{B}(k+1) = \mathbf{B}(k) + \mu_k \left[\mathbf{B}^{-T}(k) - \psi(\vec{y}(k)) \vec{x}^T(k) \right] \tag{5}$$

It can be found out that \mathbf{B}^{-T} is presented in the regulating formula when the coefficients are adjusted by random gradient algorithm. Therefore, it is necessary to calculate the inversion of the matrix \mathbf{B} . But its calculation is always very complicated, so an improved natural gradient algorithm is introduced. The improved algorithm does not have to calculate the inversion of the matrix \mathbf{B} and can accelerate the convergence. Thus the regulating formula becomes:

$$\mathbf{B}(k+1) = \mathbf{B}(k) + \mu_k \left[\mathbf{I} - \psi(\vec{y}(k)) \vec{y}^T(k) \right] \mathbf{B}(k) \tag{6}$$

Note that the above-mentioned algorithm can only be applied to super-Gaussian signals. In order to make the algorithm have the same effect on sub-Gaussian signals, an extended infomax algorithm is deduced. Its regulating formula is as follows:

$$\Delta \mathbf{B}(k) = \mu_k \left[\mathbf{I} - K \tanh(\vec{y}(k)) \vec{y}^T(k) - \vec{y}(k) \vec{y}^T(k) \right] \mathbf{B}(k) \tag{7}$$

Here, $K = \text{Diag}[\text{sgn } k_4^{(1)}, \dots, \text{sgn } k_4^{(N)}]$. In the case of super-Gaussian signals, $k_4^{(i)} > 0$, $\text{sgn } k_4^{(i)} = +1$; and in the case of sub-Gaussian signals, $k_4^{(i)} < 0$, $\text{sgn } k_4^{(i)} = -1$. $k_4^{(i)}$ can be estimated by the following expression:

$$k_4^{(i)} = E(\sec h^2 y_i)E(y_i^2) - E[(\tanh y_i) y_i] \quad (7)$$

4 Results

The nineteen-channel EEG data are decomposed into nineteen independent components by the adaptive infomax algorithm that has been mentioned above. The temporal waveforms of the nineteen independent components are shown in figure 3.

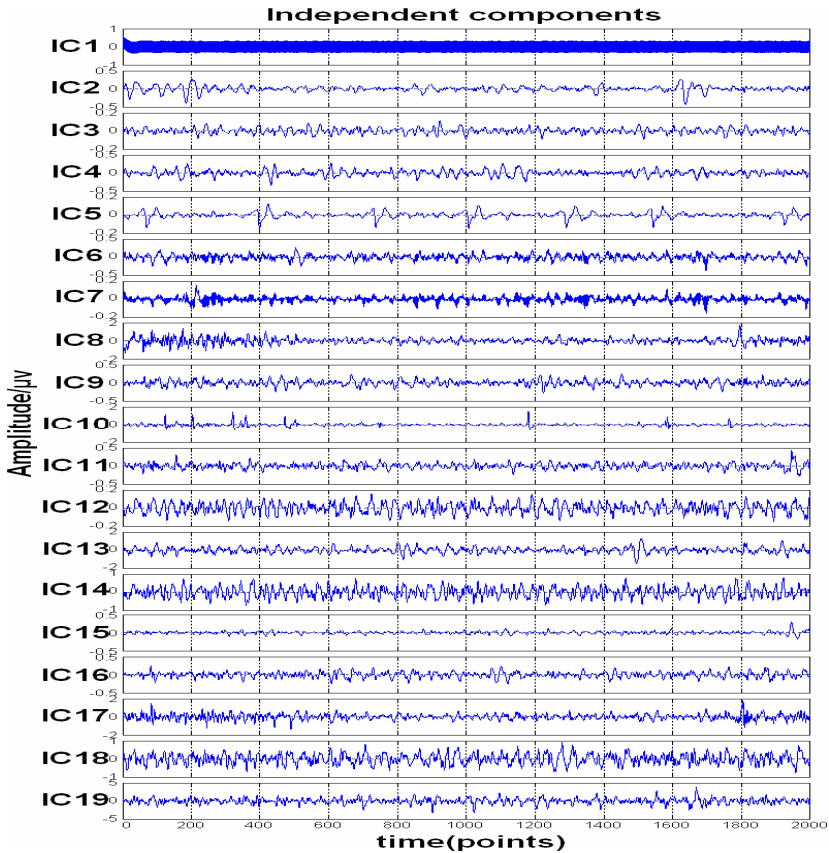


Fig. 3. Nineteen independent components decomposed by adaptive infomax algorithm

The independent components can be analyzed by three nonlinear statistical parameters in order to find out the components that are artifacts. These parameters are expounded as follows:

(1) ApEn: The nonlinear dynamic parameter --approximate entropy (ApEn) can be used to measure the complexity of the time series. This parameter can be used to assess the complexity of the independent components, then by the initialization of the threshold the components that are periodical can be found out.

(2) Hurst index: It is necessary to calculate the Hurst index of every independent component and the mean Hurst index of all the independent components. Afterwards, the threshold should be set by the deviation from the mean. Finally, the components that are artifacts can be chose out according to this threshold.

(3) Skewness: The skewness of every independent component need to be calculated, then the threshold of this parameter can be set. Accordingly, the artifacts can be selected out.

The results of three statistical parameters of the nineteen independent components are shown in table 1. A system should be periodical if its complexity parameter--ApEn equal zero. In this paper, the threshold is set to be 0.1. That is, if the ApEn of a signal is less than 0.1, the signal should be of periodicity^[8]. In table 1, it can be found out that the ApEn of IC1 is less than 0.1. Therefore, IC1 is a periodical signal. For the Hurst index, its definition is:

$$H = \log(R/S) / \log(CT) \quad (8)$$

In this expression, R is the difference between the maximal cumulate deflection and the minimal cumulate deflection of the time series, S is the variance of the time series, C is the constant (here C=0.5), and T is the sample number of the time series. Then H should be the Hurst index, and its value ranges from 0 to 1. Afterwards, the threshold can be set by the deviation from the mean Hurst index of all the independent components compared with the Hurst index of every independent component. In this article, the threshold is set to be 15%. If the deviation of an independent component is larger than 15%, this component ought to be artifacts. In table 1, it can be found out that the deviations of IC1 and IC2 are respectively 40% and 18%. Therefore, IC1 and IC2 are artifacts. Finally, the skewness of every the independent components is calculated. Skewness is the third-order cumulant, its definition is:

$$y = E(x - \mu)^3 / \sigma^3 \quad (9)$$

In this expression, μ is the mean of the time series; σ is the standard deviation of the time series. Here, the threshold of the skewness ranges from -0.3 to 0.3. That is, an independent component can be considered as artifacts if its skewness is less than -0.3 or larger than 0.3. Accordingly, four independent components IC2, IC5, IC10 and IC15 can be detected, their skewnesses are respectively -0.8376, -0.4980, 2.5117 and 0.6313, so they are all artifacts. Finally, by the three statistical parameters, five independent components IC1, IC2, IC5, IC10 and IC15 are selected out and can be ensured of artifacts.

Table 1. Three nonlinear parameters of nineteen independent components

Independent components	ApEn	Hurst	Skewness
IC1	0.0012	0.3144	1.4423e-004
IC2	0.8899	0.6140	-0.8376
IC3	0.8871	0.4897	0.0802
IC4	1.2587	0.5263	0.0357
IC5	0.7300	0.5666	-0.4980
IC6	1.3768	0.4988	-0.2330
IC7	1.1992	0.5114	-0.2024
IC8	1.2237	0.5779	0.1710
IC9	1.3644	0.5481	-0.0301
IC10	0.7461	0.5243	2.5117
IC11	1.0749	0.5665	0.1360
IC12	1.0475	0.5124	-0.0325
IC13	1.2256	0.5553	-0.0092
IC14	1.0176	0.4986	0.0933
IC15	0.8932	0.5772	0.6313
IC16	1.0221	0.4945	-0.1423
IC17	1.2589	0.5013	0.1127
IC18	1.2296	0.4897	0.0316
IC19	1.1269	0.5263	-0.1420

According to the three nonlinear statistical parameters, different kinds of artifacts are identified. This course, which does not need human interference, is a completely automatic method for artifacts removal. Some thresholds need to be set by the analysis of the statistical parameters. But these thresholds always change with different people and do not have a uniform standard. Therefore, the further studies of the thresholds need to be done in the future.

After the analysis of parameters for all the independent components, the artifacts can be detected. The spectrum and EEG topography of all the artifacts are shown in figure 4, from which the types of artifacts can be determined. The EEG topography indicates the brain spatial distribution of the independent component^[9]. By the enlargement of the x-coordinate for the temporal waveform of IC1, it can be found out that IC1 is a periodical triangular wave. And from its spectrum it can be detected that the frequency is 50Hz. So it can be said that IC1 is a high-frequency artifacts. Additionally, its spatial distribution focuses on central-parietal lobe and left frontal-temporal lobe. By the analysis of the spectrum for IC2, it can be detected that its

frequency ranges from 0 to 1Hz. In addition, from the EEG topography of IC2 it can be found out that the signal mainly focuses on the frontal lobe or circumocular. The above-mentioned aspects both accord with the properties of electrooculography (EOG) signals, so it can be said that IC2 is the eye artifacts caused by eye blinking. Similarly, by the EEG topography of IC5, IC10 and IC15, it can be found out that IC5 mostly focuses on the left temporal lobe that accords with the property of electromyography (EMG) signals; IC10 focuses on the occipital lobe and has a peak spectrum near by 10Hz, so IC10 belongs to the alpha band contained in the EEG signals; then the region of activity (ROA) for IC15 mainly appears on the contour line of the brain, and there is a rapid descending nearby the 1950th sample point of the time series which is thought to be caused by a instantaneous movement of head, therefore IC15 is a head movement artifacts.

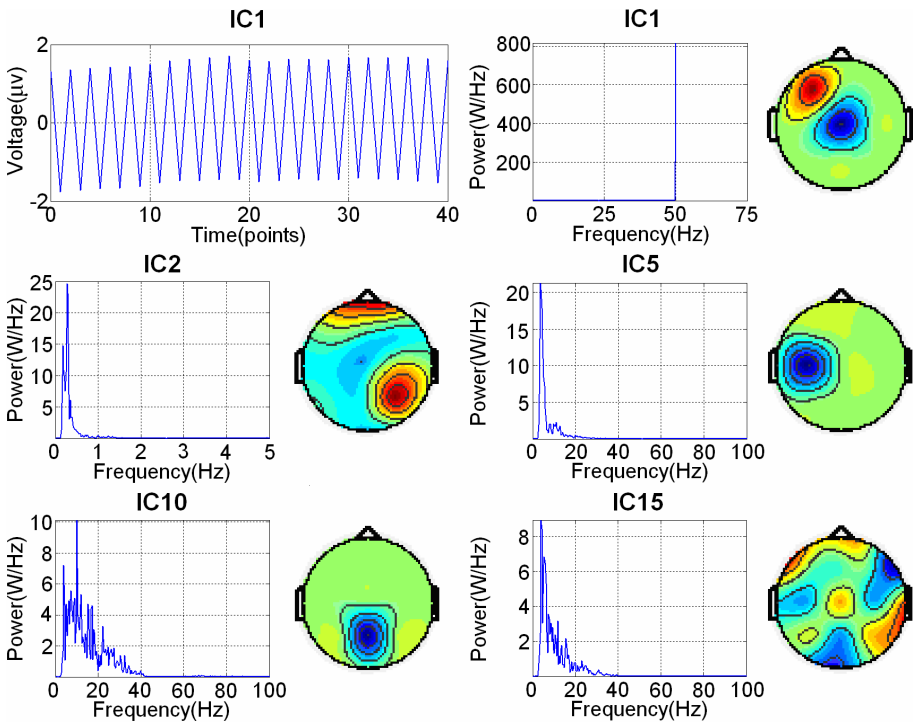


Fig. 4. Spectrum and EEG topography of artifacts

Setting all the artificial components to be zero, and projecting the other components to the scalp electrodes, then the purer Electroencephalograms can be gained. The reconstructed nineteen-channel EEG signals are shown in figure 5. In figure 5, it can be found out that the various artifacts are separated from the EEGs successfully with the use of adaptive Infomax algorithm and removal of artifacts is realized by signal reconstruction. Therefore, adaptive Infomax algorithm is thought to be a potential tool in removal of artifacts in mixed physiological signals.

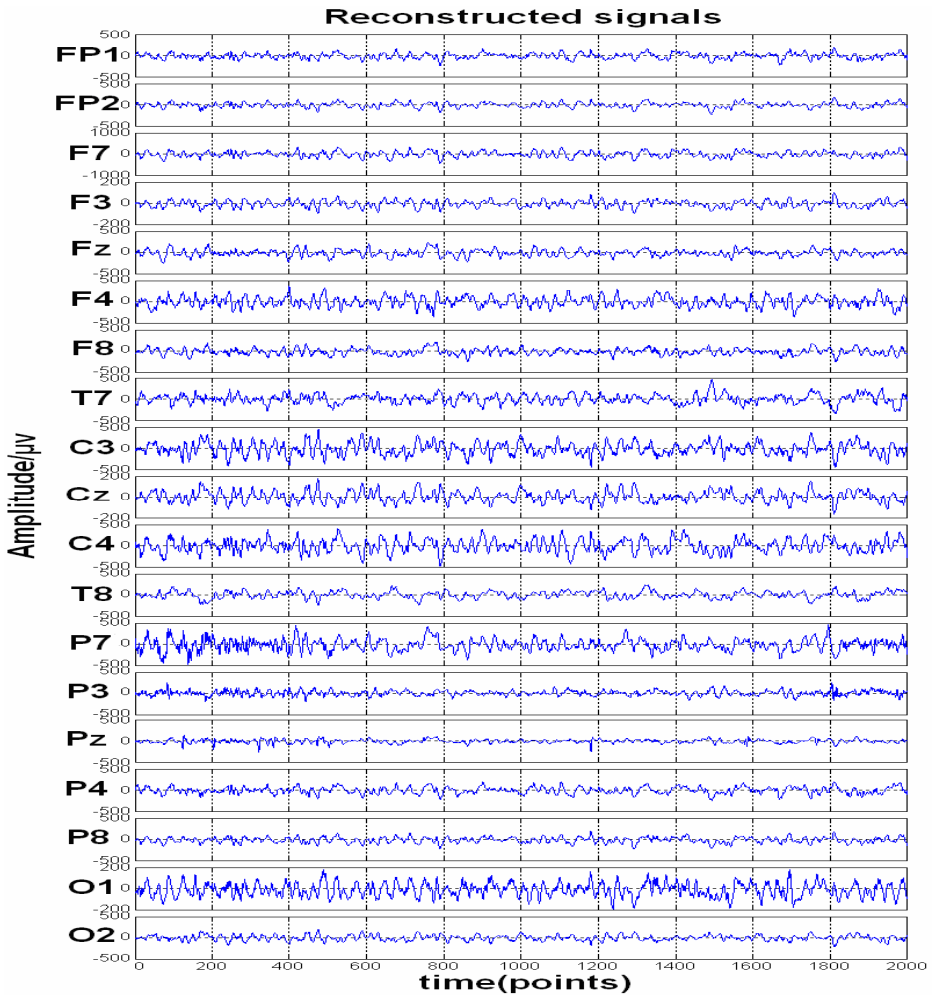


Fig. 5. Reconstructed EEG signals after the artifacts removal

5 Conclusions

The EEG signals with the magnitude of μV are always influenced by some artifacts, for example, the power frequency artifacts, eye blink artifacts, eyeball movement, electrocardiogram (ECG), electromyography (EMG) and so on. The problem that the researchers have to confront with is how to acquire the properties of the nerve and to remove the artifacts.

The reason for using BSS to remove artifacts in EEG signals is that the brain activity and the other sources (for example, EOG, ECG and so on) are different physiological courses. That is, they are statistical independent with one another which accords with the precondition of BSS. If the noise has been confirmed and can be

produced artificially or the artifacts are of physiological meanings, that is, the artifacts and the useful physiological signals can be thought of producing by independent sources, the noise or artifacts can be decomposed into independent source signals. The EEG signals not only contain super-Gaussian components, but also contain sub-Gaussian components, so it is a reasonable choosing to apply the adaptive infomax algorithm for the analysis of EEG signals.

Acknowledgment

This work was supported by National Natural Science Foundation of China under grant NO. 60371023.

References

1. Olsson, R.K., Hansen, L.K.: Linear State-Space Models for Blind Source Separation. *Journal of Machine Learning Research* 7, 2585–2602 (2006)
2. Jeong, W.J., Tae, S.K., Sung, H.K., Manbir, S.: Application of Independent Component Analysis with Mixture Density Model to Localize Brain Alpha Activity in fMRI and EEG. *International Journal of Imaging Systems and Technology* 14, 170–180 (2004)
3. Jung, T.P., Makeig, S., Humphries, C., et al.: Removing Electroencephalographic Artifacts by Blind Source Separation. *Psychophysiology* 37, 163–178 (2000)
4. Nazareth, P.C., Valeri, A.M.: Recovering EEG Brain Signals: Artifact Suppression with Wavelet Enhanced Independent Component Analysis. *Journal of Neuroscience Methods* 158, 300–312 (2006)
5. <http://www.bsp.brain.riken.jp/ICALAB/ICALABSignalProc/benchmarks/>
6. Asadi, M., Ebrahimi, N., Hamedani, G.G., Soofi, E.S.: Maximum Dynamic Entropy Models. *Journal of Applied Probability* 41, 379–390 (2004)
7. Gupta, M.R., Gray, R.M., Olshen, R.A.: Nonparametric Supervised Learning by Linear Interpolation with Maximum Entropy. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 28, 766–781 (2006)
8. Homero, R., Aboy, M., Abásolo, D., Mcnames, J., Goldstein, B.: Interpretation of Approximate Entropy: Analysis of Intracranial Pressure Approximate Entropy During Acute Intracranial Hypertension. *IEEE Transactions on Biomedical Engineering* 52, 1671–1680 (2005)
9. Hyvarinen, A., Hoyer, P.O., Inki, M.: Topographic Independent Component Analysis. *Neural Computation* 13, 1525–1529 (2001)

A Hybrid VNS with TS for the Single Machine Scheduling Problem to Minimize the Sum of Weighted Tardiness of Jobs

Xianpeng Wang and Lixin Tang

The Logistics Institute, Northeastern University, Shenyang, China
wangxianpeng@ise.neu.edu.cn,
qhjytlx@mail.neu.edu.cn

Abstract. This paper presents a hybrid variable neighborhood search (VNS) with tabu search for the single machine scheduling problem to minimize the sum of weighted tardiness of jobs, a well-known strongly NP-hard problem. In this algorithm, a pool of many solutions is used in the VNS to simultaneously generate multiple trial solutions to improve the search diversification and a tabu search is incorporated in the local search procedure of VNS to improve the search intensification. Computational results show that the proposed VNS algorithm can obtain good solutions for all standard benchmark problems.

1 Introduction

Single machine scheduling is the process of assigning a group of tasks to a single machine. The tasks are arranged so that one or many performance measures may be optimized. The performance measures of the tasks in the single machine scheduling problem include: Tardiness, Earliness, Lateness, Flowtime. Many solution techniques have been applied to solving single machine scheduling problems. Some of them are listed below. (1) Shortest Processing Time (SPT). The SPT schedule is optimal if the objective is to minimize the average flowtime. (2) Earliest Due Date (EDD). The EDD schedule is optimal if the objective is to minimize the maximum lateness. In this paper, we consider the single machine scheduling to minimize the sum of weighted tardiness of all jobs, which is strongly NP-hard [1].

Many researchers have studied this problem and proposed many approaches. A survey of dynamic programming and branch and bound algorithms [2] showed that these exact algorithms are computationally inefficient when the number of jobs is beyond 50. Therefore, many researchers turn to develop heuristics to obtain near optimal schedules in reasonable time. Several construction heuristics and dispatching rules were reviewed in [3-4]. Many other complex metaheuristics are also proposed for this problem such as genetic algorithm [5-6], simulated annealing [3], and tabu search [7]. The best algorithm for this problem in the literature is the iterated dynasearch [8]. Though the variable neighborhood search (VNS) has been successfully applied in many combinatorial optimization problems [9], existing research shows a lack of discussion of VNS for this problem. Therefore, in this paper we present an improved version of VNS for this problem.

2 Hybrid Variable Neighborhood Search with Tabu Search

The VNS is a simple and effective metaheuristic recently proposed by [10]. Different from most local search heuristics, the principle of VNS is to use two or more neighborhoods and systematically change the neighborhood within a local search algorithm.

To further improve the performance of the basic VNS, we propose the following three improvements. (1) A pool with many solutions taking into account the solution quality and diversity is introduced. (2) Multiple trial solutions x' are simultaneously generated in the *Shaking* procedure to help the search to both expand the exploration space and escape being trapped in local optimum. (3) A tabu search is adopted in the local search procedure to improve the search intensification. The general framework of our proposed IVNS can be described as follows.

- Step 1.* Select the set of neighborhood structures N_k ($k=1, \dots, k_{max}$) that will be used in the search. Initialize the solution pool P with N_{pop} solutions and set $i=1$.
- Step 2.* Repeat the following steps if $i \leq I_{max}$ (the maximum number of iterations):
- (1) If the best solution found so far has not been improved for a given number of successive iterations, re-initialize the solution pool P ;
 - (2) Set $k=1$;
 - (3) Repeat the following steps if $k \leq k_{max}$:
 - *Shaking*: Generate a random number m ($1 \leq m \leq 5$), select m solutions at random from P , and use a *solution combination method* to transform these selected solutions into a new solution x . Then randomly generate n_{trial} solutions x' from the k -th neighborhood of x .
 - *Local search*: For each x' , apply a tabu search on x' . Denote the so obtained best local optimum from all these trial solutions as x'' .
 - *Move or not*: If x'' is better than the best solution found so far, update the best solution found, and set $k=1$; otherwise set $k=k+1$.
 - *Update the pool*: If x'' is better than the best solution found so far, replace the worst solution in P with x'' ; otherwise select a solution that is farthest from the best one in P according to some distance definition for solutions and replace this selected solution with x'' .
 - (4) Set $i=i+1$.

2.1 Solution Pool Initialization and Update Methods

The solution pool P with N_{pop} (generally $N_{pop} \geq 10$) solutions is initialized using two types of heuristic procedures.

(1) Deterministic rules: Let p_i denote the processing time and w_i the due date of job i . The first four solutions in P are generated using four different deterministic priority rules: the weighted shortest processing time (WSPT) rule sequences jobs in order of non-decreasing p_i/w_i ; the EDD rule generates a job sequence in order of non-decreasing due dates; the weighted due date (WDD) rule generates a job sequence in order of non-decreasing d_i/w_i ; the NEH rule creates a job sequence as follows:

- Step 1.* Create the job sequence s using the WDD rule. Store this sequence and set the initial solution to be empty.

Step 2. Take the first two jobs in the sequence s , insert them into the initial solution, and determine their sequence to minimize the objective function as if the problem only consists of these two jobs. Delete these two jobs from s .

Step 3. Take the first job in s and insert it into the previously obtained partial initial solution at the position that gives the least increased objective function value. Delete the job from s . Repeat this step until a complete initial solution is obtained.

(2) Randomized rule: To improve the diversity of P , we use a randomized rule based on the above NEH rule to initialize other solutions in P . Different from NEH in step 3, this method takes a job at random from the first three jobs of s . Therefore, each run of this method will randomly generate a different initial solution for P .

During the search process of IVNS, if the best local optimum x'' is better than the best solution found so far, then the worst solution in P will be replaced by x'' . Otherwise, for this problem we first define a distance between two solutions $s_1=(a_1, \dots, a_n)$, $s_2=(b_1, \dots, b_n)$ as $Dist(s_1, s_2) = \sum_{i=1}^n sign(|a_i - b_i|)$ in which $sign(s)=1$ if $s \neq 0$; otherwise $sign(s)=0$. Then we select a solution that is farthest from the best one in the pool and replace this selected solution with x'' .

2.2 Neighborhoods

Three kinds of neighborhoods are adopted in the IVNS based on three kinds of *moves*: (1) *ForeInsertion*: remove a job from its current position a ($3 \leq a \leq n$) in the solution and reinsert it at a former position b ($1 \leq b \leq a-2$); (2) *BackInsertion*: remove a job from its current position a ($1 \leq a \leq n-2$) in the solution and reinsert it at a latter position b ($a+2 \leq b \leq n$); and (3) *Swap*: swap two jobs at two positions.

2.3 Solution Combination Method

Let S_j denote the j -th solution in the set of m solutions and $f(S_j)$ the corresponding objective function value. Let $z_{ikj}=1$ if job i is arranged at position k in S_j , otherwise $z_{ikj}=0$. Then the combination method used in our IVNS can be described as follows.

Step 1. Calculate the relative value v_j of each S_j by $v'_j = \frac{f(S_1)}{f(S_j)}$

and $v_j = v'_j / \sum_{l=1}^m v'_l$, then the weight value of each job i ($i=1, \dots, n$) at

position k can be calculated by $R_{ik} = \sum_{j=1}^m z_{ikj} v_j$.

Step 2. Set $k=1$. Select the job with the highest weight value at position k and then arranged it at the position k in the new solution. Set $k=k+1$ and repeat this step until $k=n$. Note that the same job may be repetitively arranged at different positions if it has highest weight values for different positions.

Step 3. Remove the reduplicate jobs arranged in the new solution. That is, if job i is arranged in positions a_1, a_2 , and a_3 ($a_1 < a_2 < a_3$), then remove job i at positions a_2 and a_3 . And then insert the unarranged jobs into the new solution with the least increased objective function value.

2.4 Tabu Search

Tabu search (TS) is a memory based metaheuristic that can avoid being trapped in local optimum by allowing non-improving moves and can prevent cycling back to previous visited solutions by a tabu list which records recent adopted moves. The tabu search used in our IVNS follows the standard framework proposed by [11][12].

Neighborhood. At each iteration of our TS, the three moves *ForeInsertion*, *BackInsertion*, and *Swap* are all performed, and the best move is selected. If the best move is tabooed, select the best one from other neighborhoods. If the selected best move is still tabooed, then perform a 4-cycle-shift move, which change a solution $S=(s(1), \dots, s(i), s(i+1), \dots, s(j), s(j+1), \dots, s(n))$ to be $S'=(s(1), \dots, s(j+1), s(j), \dots, s(i), s(i+1), \dots, s(n))$. This kind of move cannot be easily restored by either a move from the three moves on the current solution.

Tabu list. Once a move is accepted and performed, its reverse move will be recorded in the tabu list. The number of iterations that the recorded move should be kept in the tabu list is called the tabu list length. In our TS, the tabu list length is set to a fixed number 7.

Stopping criteria. Our TS terminates when the maximum iteration is reached.

2.5 Speedups

Due to the large number of candidate moves in the local search procedure, the evaluation of each move that will be performed will cost a great deal of time. Therefore, in our algorithm two kinds of speedups are used.

(1) The first speedup takes use of elimination rules to avoid evaluating inferior moves based on two corollaries proposed in [13]. Given a solution $S=(s(1), \dots, s(j), \dots, s(i), \dots, s(n))$, take the job $s(i)$ and consider all *Foreinsertion* and *Swap* moves with the job $s(j)$. If at least one of the following conditions $d_{s(j)} > d_{s(i)}$, or $w_{s(j)} < w_{s(i)}$, or $p_{s(j)} > p_{s(i)}$ is satisfied, then job $s(j)$ does not have to precede job $s(i)$ and job $s(i)$ can thus be inserted before job $s(j)$ or swapped with job $s(j)$. We call this kind of moves as *acceptable moves*. Furthermore, one can easily find that if $C_{s(i)} < d_{s(i)}$, the moves that insert job $s(i)$ before job $s(j)$ or swapped job $s(i)$ with job $s(j)$ will not give any improvement, and thus these moves can also be prevented.

(2) The second one uses programming tricks to avoid calculating the weighted tardiness of each job of a solution generated by an acceptable move. In the following, we take the evaluation of the *Swap* move to describe this kind of speedup. Given a solution $S=(s(1), \dots, s(n))$, we first calculate the completion time $C_{s(i)}$ and the weighted tardiness $T_{s(i)}$ of each job $s(i)$. Then for each acceptable *Swap* move that swaps job $s(i)$ with job $s(j)$ ($i < j$), the decreased objective function value obtained by performing this move can be calculated by:

$$w_{s(i)} (C_{s(j)} - d_{s(i)})^+ + w_{s(j)} (C_{s(i-1)} + p_{s(j)} - d_{s(j)})^+ + \sum_{q=i+1}^{j-1} w_{s(q)} (C_{s(q)} + p_{s(j)} - p_{s(i)} - d_{s(q)})^+ - \sum_{q=i}^j T_{s(q)},$$

where the first and the second item are the weighted tardiness of job $s(i)$ and job $s(j)$ in the new solution respectively, the third item is the total weighted tardiness of jobs

from position $i+1$ to $j-1$ in the new solution, and the last item is the total weighted tardiness of jobs from position i to j in the original solution S . Using this speedup, the evaluation of a move can be restricted to those jobs that are affected by this move, and thus avoid calculating the weighted tardiness of each job in the new solution.

3 Computational Experiments

To test the performance of our IVNS algorithm, the computational experiments were carried out on a set of standard benchmark problem instances drawn from the OR-library (see <http://people.brunel.ac.uk/~mastjjb/jeb/info.html>). The benchmark set contains 375 particularly hard instances of three different sizes, selected from a large number of randomly generated problems. For each size $n=40, 50,$ and $100,$ a sample of 125 instances was provided. Our algorithm was implemented using C++, and tested on a Pentium-IV 3.0 GHz PC. In the following experiments, we perform each testing algorithm for at most 10 trial runs (if the optimal or best known solution is reached, the trial run terminates) and selected the best result.

As did by many researchers, we collect the following values for each instance: (1) *PRD*: the percentage relative deviation of a solution value f_s found by an algorithm from the optimal (or best known) solution value *OPT*, $PRD=100(f_s - OPT)/OPT$; (2) *APRD*: the average *PRD* values for a sample of 125 instances; (3) *MPRD*: the maximum *PRD* out of a sample of 125 instances; (4) *NO*: the number of optimal or best known solutions found out of a sample of 125 instances; (5) CPU_{opt} : the average computation time (in seconds) for an algorithm to find the optimal or best known solution value out of a sample of 125 instances.

3.1 Parameter Setting and Comparison with Basic VNS

The parameters used in the IVNS are set as follows: the size of pool $N_{pop}=20,$ the maximum iteration $I_{max}=1000,$ and the number of candidate solutions from the population to be combined into a new solution $m=5.$

For the number of trial solutions generated in a certain neighborhood (n_{trial}), it is found that it has significant influence on the performance of IVNS. To show the effectiveness of the introduction of solution pool, we also tested the basic VNS by setting $N_{pop}=1$ and $n_{trial}=1$ in our IVNS, noting that the basic VNS and the IVNS use the same local search procedure. The results of the different setting of n_{trial} and the comparison between the basic VNS and our proposed IVNS are given in Table 1.

Table 1. Results of the parameter tuning and comparison with basic VNS

n	basic VNS			IVNS ($n_{trial}=1$)			IVNS ($n_{trial}=5$)			
	APRD	MPRD	NO	APRD	MPRD	NO	APRD	MPRD	NO	CPU_{opt}
40	0.01	1.01	118	0.00	0.32	123	0.00	0.00	125	6.19
50	0.25	22.92	109	0.04	4.72	118	0.00	0.00	125	12.15
100	0.15	14.62	85	0.01	0.12	93	0.00	0.00	125	183.47

Based on the results shown in Table 1, the following conclusions can be made.

- (1) With the introduction of the solution pool, our proposed IVNS outperforms the basic VNS in both the solution quality and stability.
- (2) With the increase of n_{trial} , the solution quality of the IVNS improves.

3.2 Performance of Speed-Up Strategy

To show the effectiveness of the proposed speedups, we tested the IVNS without speedups ($n_{trial}=5$) and compared it with the IVNS with speedups ($n_{trial}=5$). Since the IVNS without speedups needs a relatively more time, we set $I_{max}=10$ (note that $I_{max}=1000$ for the IVNS with speedups). The comparison results are given in Table 3, from which it can be found that using the proposed speedups the computation time for IVNS to find the optimal or best known solutions can be greatly reduced.

Table 2. Comparison between IVNS without speedups and IVNS with speedups

n	IVNS (without speedups)				IVNS (with speedups)			
	APRD	MPRD	NO	CPU_{opt}	APRD	MPRD	NO	CPU_{opt}
40	0.00	0.00	125	21.85	0.00	0.00	125	6.19
50	0.00	0.00	125	79.28	0.00	0.00	125	12.15
100	0.05	4.78	113	562.75	0.00	0.00	125	183.47

3.3 Comparison with Other Metaheuristics

We further compared our IVNS ($n_{trial}=5$) with other metaheuristics that are recently published in literature. The results are given in Table 3, which show that our IVNS is comparable with the metaheuristics proposed by [6][7] in the solution quality.

Table 3. Comparison between IVNS and other metaheuristics

n	Avci et al. [6]				Bilge et al. [7]				IVNS ($n_{trial}=5$)			
	APRD	MPRD	NO	CPU	APRD	MPRD	NO	CPU_{opt}	APRD	MPRD	NO	CPU_{opt}
40	0.000	0.000	125	29.11	0.000	–	125	2.74 ^a	0.000	0.000	125	6.19
50	0.000	0.020	124	41.02	0.001	–	124	16.91 ^a	0.000	0.000	125	12.15
100	0.020	0.300	83	118.97	0.007	–	108	127.59 ^a	0.000	0.000	125	183.47

CPU – the operation time of entire search process on a Pentium II 400 MHz PC.

a – the tests were carried out on a Petinum IV 1.6 GHz PC.

4 Conclusions

In this paper, a hybrid VNS with TS algorithm is developed for solving the single machine scheduling problem to minimize the sum of weighted tardiness of jobs. In this algorithm, the search diversification and the search intensification are both improved by adopting a solution pool of many elite solutions to simultaneously

provide multiple trial solutions and a tabu search to act as the local search, respectively. Speedup tricks are also developed to reduce the computation time. The computational results show that the hybrid algorithm can obtain good solutions for all benchmark problems, compared to the basic VNS and other two metaheuristics from recently published literature. Further research will be made to develop more improvement strategies for the IVNS and extend the IVNS to other combinatorial optimization problems such as the hybrid flowshop scheduling problem.

Acknowledgements. This research is supported by National Natural Science Foundation for Distinguished Young Scholars of China (Grant No. 70425003), National 863 High-Tech Research and Development Program of China (Grant No. 2006AA04Z174), National Natural Science Foundation of China (Grant No. 60674084).

References

1. Lawler, E.L.: A Pseudopolynomial Algorithm for Sequencing Jobs to Minimize Total Tardiness. *Annals of Discrete Mathematics* 1, 331–342 (1977)
2. Abdul-Razaq, T.S., Potts, C.N., VanWassenhove, L.N.: A Survey of Algorithms for the Single Machine Total Weighted Tardiness Scheduling Problem. *Discrete Applied Mathematics* 26, 235–253 (1990)
3. Potts, C.N., Van Wassenhove, L.N.: Single Machine Tardiness Sequencing Heuristics. *IIE Transactions* 23, 346–354 (1991)
4. Volgenant, A., Teerhuis, E.: Improved Heuristics for the N-job Single Machine Weighted Tardiness problem. *Computers and Operations Research* 26(1), 35–44 (1999)
5. Crauwels, H.A.J., Potts, C.N., Van Wassenhove, L.N.: Local Search Heuristics for the Single Machine Total Weighted Tardiness Scheduling Problem. *INFORMS Journal on Computing* 10(3), 341–350 (1998)
6. Avci, S., Akturk, M.S., Storer, R.H.: A Problem Space Algorithm for Single Machine Weighted Tardiness Problem. *IIE Transactions* 35, 479–486 (2003)
7. Bilge, Ü., Kurtulan, M., Kırac, F.: A Tabu Search Algorithm for the Single Machine Total Weighted Tardiness Problem. *European Journal of Operational Research* 176, 1423–1435 (2007)
8. Congram, R.K., Potts, C.N., Van de Velde, S.L.: An Iterated Dynasearch Algorithm for the Single Machine Total Weighted Tardiness Scheduling Problem. *INFORMS Journal on Computing* 14(1), 52–67 (2002)
9. Hansen, P., Mladenović, N.: Variable Neighborhood Search: Principles and Applications. *European Journal of Operational Research* 130, 449–467 (2001)
10. Mladenović, N., Hansen, P.: Variable Neighborhood Search. *Computers and Operations Research* 24, 1097–1100 (1997)
11. Glover, F.: Future Paths for Integer Programming and Links to Artificial Intelligence. *Computers and Operations Research* 13, 533–549 (1986)
12. Glover, F.: Tabu Search-Part I. *ORSA Journal of Computing* 1(3), 190–206 (1989)
13. Rinnooy Kan, A.H.G., Lageweg, B.J., Lenstra, J.K.: Minimizing Total Costs in One-Machine Scheduling. *Operations Research* 25, 908–927 (1975)

A Wide Neighborhood Primal-Dual Interior-Point Algorithm for a Class of Convex Programming*

Yuqin Zhao and Mingwang Zhang**

College of Science, China Three Gorges University,
Yichang, Hubei, 443002, China
zmwang@ctgu.edu.cn

Abstract. This paper presents a new primal-dual interior-point algorithm with reduced potential function for a class of convex programming, based on the ideas of that method for solving linear programming. The new algorithm chooses the classical Newton direction as iteration direction and its iteration stepsize is determined by potential function. As the search directions Δx and Δs aren't orthogonal any more, the complexity analysis of this method is different from that of linear programming, correspondingly. Under a scaled Lipschitz condition, the algorithm is proved to possess $O(nL)$ iteration-complexity bounds

Keywords: Wide-neighborhood interior-point algorithm; Potential function; Scaled Lipschitz condition; Iteration complexity.

1 Introduction

In 1984, Karmarkar introduced a new interior-point algorithm for linear programming in [4]. Since it can compete with the Simplex method for being more efficient in worst-case complexity analysis and computation of medium and large-scale problems, it rapidly attracted broad attention of academia. From then on, Linear Optimization (LO) became an active area of research. The resulting Interior-point Methods (IPMs) are now among the most effective methods for solving LO problems.

Compared with the Simplex method, an important advantages of IPMs is that IPMs extend gracefully to nonlinear convex optimization, whereas the Simplex method does not. This observation was made first in 1988 [13], by Nesterov and Nemirovsky, who developed a general framework for nonlinear convex optimization problems [14,15]. They proved that IPMs can be extended to solve convex programs in principle based on the notion of self-concordant functions. Particularly, They show that linear programs, convex quadratic programs with convex quadratic constraints and semidefinite programs all have explicit and easily computable self-concordant barrier functions, and hence can be solved in polynomial time.

* Supported by Natural Science Foundation of Educational Commission of Hubei Province of China (NO. D200613009).

** Corresponding author.

Some attentions have been devoted to study convex programs, which possess extensive applications in various areas, such as control area. In the late 70s, Richalet [2] developed model predictive control, in which linear or quadratic programs are used to solve an optimal control problem at each time step. [9] discussed three examples of areas of control such as robust open-loop optimal control, and so on. For these backgrounds, numerous researchers study methods for solving them, such as [3,5,6,7]. D.Hertog [1] introduced a sufficient condition for self-concordance of the logarithmic barrier function associated with those programs which have the polynomial complexity results. The monographs [10,11] give a systematic summarize on the results of interior-point algorithms for solving convex programming.

The goal of this paper is to establish a new interior-point algorithm for a class of convex programming, that is, wide neighborhood primal-dual interior-point algorithm based on reduced potential function, on the basis of the ideas of that method for solving LO problems in [8]. After presenting the fundamental expressions which differ from those of LO problems, the algorithm uses the Newton direction as iteration direction and its iteration step is determined by potential function. As the search direction Δx and Δs aren't orthogonal any more, its complexity analysis is different from that of linear programming, correspondingly. Under a scaled Lipschitz condition, the algorithm is proved to possess $O(nL)$ iteration-complexity bounds.

This paper is organized as follows. In Section 2, we introduce the problem which will be the subject of our study with some basic concepts underlying the algorithm and present the corresponding assumptions that will be used in our presentation. In Section 3, we introduce how the algorithm generate the direction of movement at each iteration and how they compute the next iterate using this direction. Then we describe the potential function with its properties for the stepsize. And we present the algorithm at the end of the section. In Section 4, the complexity analysis of the algorithm is made in the form of theorems and the correspondent proofs are offered. Some conclusions are given in Section 5.

2 Preliminaries

Throughout this paper, we deal with the following standard linearly constrained convex programming:

$$(CP) \quad \min \{f(x) : Ax = b, x \geq 0\},$$

where $f(x)$ is a twice continuously differentiable convex function, $A \in R^{m \times n}$ is a real $m \times n$ matrix with rank $A = m$ and $b \in R^m$ are the datas, $x \in R^n$ is the primal variable. The corresponding dual problem is given by

$$(DP) \quad \max \{f(x) - \nabla f(x)^T x + b^T y : \nabla f(x) + A^T y + s = 0, s \geq 0\},$$

where $y \in R^m$ and $s \in R^n$ are the dual variables.

We define the sets of primal-dual feasible interior-points as following

$$S^0 = \{(x, y, s) : Ax = b, -\nabla f(x) + A^T y + s = 0, x > 0, s > 0\},$$

and the neighborhood of the central path, where our sequences of iterates as

$$\mathcal{N}_\infty^-(\beta) = \left\{ (x, s) \in S^0 \mid Xs \geq (1 - \beta)\mu e, \text{ where } \mu = \frac{x^T s}{n}, \beta \in (0, 1) \right\}.$$

It's not difficult to discover the following inclusion holds.

$$\mathcal{N}_2(\beta) \subset \mathcal{N}_\infty(\beta) \subset \mathcal{N}_\infty^-(\beta) \subset S^0, \quad \text{for each } \beta \in (0, 1),$$

and when β is close to 1, the neighborhood $\mathcal{N}_\infty^-(\beta)$ spreads almost all over the feasible region. So it is regarded as a wide neighborhood of the central path whereas the neighborhood $\mathcal{N}_2(\beta)$ is a narrow one.

According to the dual theory, (x, y, s) is an optimal solution to (CP) and (DP) if and only if it satisfies the following optimality system:

$$Ax = b, \quad x \geq 0, \tag{1}$$

$$-\nabla f(x) + A^T y + s = 0, \quad s \geq 0, \tag{2}$$

$$x^T s = 0. \tag{3}$$

We use the following notational conventions throughout the paper. R_{++}^n denotes the set of the positive real numbers, x denotes the vector of $(x_1, \dots, x_n)^T$, $X = \text{diag}(x)$ denotes the diagonal matrix with the components of the vector on the diagonal, and e denotes the vector of ones in R^n . $\|\cdot\|$ and $\|\cdot\|_\infty$ expresses the l_2 -norm and l_∞ -norm, respectively.

Assumption 1. *We make the following assumptions throughout our paper.*

- (I) *the feasible interior-point set S^0 is non-null;*
- (II) $\forall x \in S^0$, $f(x)$ *satisfies a scaled Lipschitz condition, which will be defined in the sequel.*

Definition 1. Scaled Lipschitz Condition *There exists $M \geq 1$, such that*

$$\|x[\nabla f(x + \Delta x) - \nabla f(x) - \nabla^2 f(x)\Delta x]\|_\infty \leq M |\Delta x^T \nabla^2 f(x)\Delta x|,$$

whenever $x > 0$ and $\|X^{-1}\Delta x\| \leq 1$.

3 Descriptions of the Algorithm

The basic idea of the new algorithm is: taking along the Newton direction $(\Delta x, \Delta y, \Delta s)$ a step with the step-size determined by the potential function, one generates a sequence of iterates in the neighborhood $\mathcal{N}_\infty^-(\beta)$ of central path, then the limit yields to the optimal solutions for (CP) and (DP).

Solving the following Newton system at the current approximate solution $(x, y, s) \in \mathcal{N}_\infty^-(\beta)$:

$$\begin{aligned} A\Delta x &= 0, \\ -\nabla^2 f(x) + A^T \Delta y + \Delta s &= 0, \\ S\Delta x + X\Delta s &= \gamma\mu e - Xs, \end{aligned} \tag{4}$$

where $\gamma \in (0, 1)$, $\beta \in (0, \frac{2}{3})$ are given constants, and $\gamma \leq 1 - \frac{\beta}{2}$, $\mu = \frac{x^T s}{n}$, one obtains the search direction $d = (\Delta x, \Delta y, \Delta s)$, then gets a new triple along it by moving a stepsize θ in the following format:

$$x(\theta) = x + \theta \Delta x; \quad y(\theta) = y + \theta \Delta y; \quad s(\theta) = \nabla f(x(\theta)) - A^T y(\theta).$$

Remark 1. One can easily deduce that $s(\theta) = s + \theta \Delta s + \nabla f(x + \theta \Delta x) - \nabla f(x) - \theta \nabla^2 f(x) \Delta x$. If denotes $t(\theta) = \nabla f(x + \theta \Delta x) - \nabla f(x) - \theta \nabla^2 f(x) \Delta x$, then $s(\theta)$ can be simplified as $s(\theta) = s + \theta \Delta s + t(\theta)$.

Remark 2. Note that $\Delta x^T \Delta s = \Delta x^T \nabla^2 f(x) \Delta x \geq 0$, which is the key point where the new algorithm is different from that of [8] for LO programs.

Before stating how to select the stepsize $\bar{\theta}$, which will be described in the algorithm frame, we will first introduce the potential function.

Definition 2. Potential Function. Let $x, s \in R_{++}^n$, we define

$$\Psi(x, s) = \rho \log(x^T s) - \sum_{i=1}^n \log(x_i s_i)$$

as the primal-dual potential function at (x, s) , where $\rho > n$ is a given constant.

Remark 3. The primal-dual potential function can be written as

$$\Psi(x, s) = (\rho - n) \log \mu + \rho \log n - \sum_{i=1}^n \log \frac{x_i s_i}{\mu}.$$

It is an important concept of IPMs, which plays a key role in complexity analysis for iterative algorithm, and possesses the following property.

Proposition 1. [8] $\forall x, s \in R_{++}^n$,

$$\Psi(x, s) \geq (\rho - n) \log(x^T s) + n \log n \geq (\rho - n) \log(x^T s).$$

Remark 4. According to this property, one will deduce $x^T s \leq \varepsilon$ when $\Psi(x, s)$ is reduced to $(\rho - n) \log \varepsilon$. This property will be used in the complexity analysis.

Algorithm 1

Input:

Proximity parameters $\beta \in (0, \frac{2}{3})$ and $\gamma \leq 1 - \frac{\beta}{2}$, an accuracy parameter $\varepsilon > 0$;
 $(x^0, y^0, s^0) \in \mathcal{N}_{\infty}^{-}(\gamma)$ satisfies $(x^0)^T s^0 > \varepsilon$;

begin

Let $(x, s) = (x^k, s^k), k = 0$;

while $(x^k)^T s^k \geq \varepsilon$ **do**

Solve (4) and compute the maximum step size $\bar{\theta}$ such that

$(x(\bar{\theta}), s(\bar{\theta})) \in \mathcal{N}_{\infty}^{-}(\beta)$, and $\Psi(x(\bar{\theta}), s(\bar{\theta})) \leq \Psi(x(\theta), s(\theta))$

for each θ with $(x(\theta), s(\theta)) \in \mathcal{N}_{\infty}^{-}(\beta)$;

Set $(x^{k+1}, s^{k+1}) = (x(\bar{\theta}), s(\bar{\theta}))$, $k = k + 1$.

end

end

4 Complexity Analysis of the Algorithm

Before bounds on $\bar{\theta}$ in detail, we first note that the following relations hold.

$$X(\theta)s(\theta) = (1 - \theta)Xs + \theta\gamma\mu e + \theta^2 \Delta X \Delta s + X(\theta)t(\theta), \tag{5}$$

$$\mu(\theta) = (1 - \theta)\mu + \theta\gamma\mu + \frac{\theta^2}{n} \Delta x^T \Delta s + \frac{x(\theta)^T t(\theta)}{n}. \tag{6}$$

If denotes $g(\theta) = \theta^2 \Delta X \Delta s + X(\theta)t(\theta)$, then last equations can be simplified as

$$X(\theta)s(\theta) = (1 - \theta)Xs + \theta\gamma\mu e + g(\theta), \quad \mu(\theta) = (1 - \theta)\mu + \theta\gamma\mu + \frac{1}{n}e^T g(\theta).$$

The following lemma can be easily gained by direct derivation .

Lemma 1. *The iterative sequence $\{(x^k, y^k, s^k)\}$, which is generated by the algorithm satisfies the feasibility of both the primal and dual problem.*

Lemma 2. *Let $p = X^{-0.5}S^{0.5} \Delta x$, $q = X^{0.5}S^{-0.5} \Delta s$, $r = (XS)^{-0.5}(\gamma\mu e - Xs)$, then the expressions $p + q = r$, $p^T q \leq \frac{1}{4} \| r \|^2$ and $\| Pq \| \leq \frac{1}{4} \| r \|^2$ hold.*

To use Lemma 2 we also need to bound r . The following result is useful.

Lemma 3. *Let r be as above. If $\beta \in (0, \frac{2}{3})$, $\gamma \leq 1 - \frac{\beta}{2}$, and $(x, s) \in \mathcal{N}_\infty^-(\beta)$, then $\| r \|^2 \leq n\mu$.*

Proof. According to the definition of Euclidean norm, one can easily obtain that

$$\| r \|^2 = \sum_{i=1}^n \frac{(\gamma\mu - x_i s_i)^2}{x_i s_i} = \sum_{i=1}^n \left(\frac{(\gamma\mu)^2}{x_i s_i} - 2\gamma\mu + x_i s_i \right).$$

Due to $(x, s) \in \mathcal{N}_\infty^-(\beta)$, which implies $x_i s_i \geq (1 - \beta)\mu$ for each i . Thus,

$$\| r \|^2 \leq \frac{n(\gamma\mu)^2}{(1 - \beta)\mu} - 2n\gamma\mu + n\mu.$$

Based on the relations $\beta \in (0, \frac{2}{3})$, $\gamma \leq 1 - \frac{\beta}{2}$, so $\gamma \leq 2(1 - \beta)$ holds. Therefore, $\| r \|^2 \leq n\mu$, which yields the final results. □

Lemma 4. *Let $\beta \in (0, \frac{2}{3})$, $\gamma \leq 1 - \frac{\beta}{2}$, $\bar{\theta}$ be the largest stepsize of the new algorithm, then*

$$\bar{\theta} \geq \theta_1 := \min \left\{ \frac{1}{\| X^{-1} \Delta x \|}, \frac{\beta\gamma}{(2 - \beta)nC^*} \right\},$$

where $C^* = \frac{1}{4} + \frac{M}{2}$, and M denotes the scaled Lipschitz constant introduced in Definition 7.

Proof. Firstly, for each $\theta \in (0, \theta_1]$, we have $\theta \| X^{-1} \Delta x \| \leq 1$. Since $f(x)$ satisfies the scaled Lipschitz condition, so there exists a constant $M \geq 1$, such that

$$\| x [\nabla f(x + \theta \Delta x) - \nabla f(x) - \theta \nabla^2 f(x) \Delta x] \|_\infty \leq \theta^2 M |\Delta x^T \nabla^2 f(x) \Delta x|.$$

As $\|I + \theta X^{-1} \Delta x\| \leq 2$, using Lemma 2 and Lemma 3 one can derive

$$\begin{aligned} \|X(\theta)t(\theta)\|_\infty &= \|(I + \theta X^{-1} \Delta x) X t(\theta)\|_\infty \leq \|I + \theta X^{-1} \Delta x\| \|X t(\theta)\|_\infty \\ &\leq 2M\theta^2 \Delta x^T \nabla^2 f(x) \Delta x = 2M\theta^2 \Delta x^T \Delta s = 2M\theta^2 p^T q \leq \frac{1}{2}M\theta^2 \|r\|^2 \leq \frac{1}{2}M\theta^2 n\mu. \end{aligned}$$

Hence, $\|g(\theta)\|_\infty = \|\theta^2 \Delta X \Delta s + X(\theta)t(\theta)\|_\infty \leq \theta^2 \|\Delta X \Delta s\|_\infty + \frac{1}{2}M\theta^2 n\mu$

$$\leq \frac{1}{4}\theta^2 \|r\|^2 + \frac{1}{2}M\theta^2 n\mu \leq \frac{1}{4}\theta^2 n\mu + \frac{1}{2}M\theta^2 n\mu.$$

Setting $C^* = \frac{1}{4} + \frac{M}{2}$, then the above relations imply $\|g(\theta)\|_\infty \leq n\mu\theta^2 C^*$. Applying $\|g(\theta)\|_\infty \leq g(\theta)_i \leq \|g(\theta)\|_\infty$, one can observe that

$$X(\theta)s(\theta) \geq (1 - \theta)Xs + \theta\gamma\mu e - \|g(\theta)\|_\infty e, \quad \mu(\theta) \leq (1 - \theta)\mu + \theta\gamma\mu + \|g(\theta)\|_\infty.$$

Thus, $X(\theta)s(\theta) - (1 - \beta)\mu(\theta)e$

$$\begin{aligned} &\geq (1 - \theta)Xs + \theta\gamma\mu e - \|g(\theta)\|_\infty e - (1 - \beta)[(1 - \theta)\mu + \theta\gamma\mu + \|g(\theta)\|_\infty] \\ &= (1 - \theta)[Xs - (1 - \beta)\mu e] + \beta\theta\gamma\mu e - (2 - \beta)\|g(\theta)\|_\infty e. \end{aligned}$$

Considering $(x, s) \in \mathcal{N}_\infty^-(\beta)$, which implies $Xs \geq (1 - \beta)\mu e$, we have

$$\begin{aligned} X(\theta)s(\theta) - (1 - \beta)\mu(\theta)e &\geq \beta\theta\gamma\mu e - (2 - \beta)\|g(\theta)\|_\infty e \\ &\geq \beta\theta\gamma\mu e - (2 - \beta)n\mu\theta^2 C^* e = (\beta\gamma - (2 - \beta)n\theta C^*)\theta\mu e \geq (\beta\gamma - (2 - \beta)n\theta_1 C^*)\theta\mu e. \end{aligned}$$

By $\theta_1(2 - \beta)nC^* \leq \beta\gamma$, so $\forall \theta \in (0, \theta_1]$, we can get

$$X(\theta)s(\theta) - (1 - \beta)\mu(\theta)e \geq (\beta\gamma - (2 - \beta)n\theta_1 C^*)\theta\mu e \geq 0.$$

Furthermore, by the strict feasibility of (x, s) , we have both $x > 0$ and $s > 0$, this together with $\theta\|X^{-1}\Delta x\|_\infty \leq 1$, we can deduce that

$$x(\theta) = x + \theta\Delta x = X(e + X^{-1}\theta\Delta x) > 0.$$

Similarly, we have $s(\theta) = s + \theta\Delta s + \nabla f(x(\theta)) - \nabla f(x) - \theta\nabla^2 f(x)\Delta x > 0$, which follows from $f : R_+^n \rightarrow R_+^n$ is continuously differentiable and convex. Summing up all these results, we get $\forall \theta \in (0, \theta_1]$, $(x(\theta), s(\theta)) \in \mathcal{N}_\infty^-(\beta)$.

Therefore, according to the definition of $\bar{\theta}$, we can obtain

$$\bar{\theta} \geq \theta_1 := \min \left\{ \frac{1}{\|X^{-1}\Delta x\|}, \frac{\beta\gamma}{(2 - \beta)nC^*} \right\},$$

which completes the proof of this lemma. □

As mentioned above, our algorithm decreases Ψ by a fixed amount depending on n each iteration. This then bounds the number of iterations required. We can now prove the following lemma.

Lemma 5. *Let β and γ be as in Lemma 4.*

$$\rho = n + \frac{3n^2(2 - \beta)C^*}{\beta\gamma \left(1 - \frac{2\gamma}{2 - \beta}\right)} \log \frac{1}{1 - \beta},$$

then we have $\Psi(x(\bar{\theta}), s(\bar{\theta})) - \Psi(x, s) \leq -2n \log \frac{1}{1 - \beta}.$

Proof. If $(x(\theta), s(\theta)) \in \mathcal{N}_{\infty}^-(\beta)$, so that $x(\theta)_i s(\theta)_i \geq (1 - \beta)\mu(\theta)$. Whence,

$$\sum_{i=1}^n \log \frac{x(\theta)_i s(\theta)_i}{\mu(\theta)} \geq n \log(1 - \beta).$$

Considering $\sum_{i=1}^n \log \frac{x_i s_i}{\mu} \leq 0$, definitions of $\bar{\theta}$ and θ_1 , using Lemma 4, we can have

$$\begin{aligned} & \Psi(x(\bar{\theta}), s(\bar{\theta})) - \Psi(x, s) \leq \Psi(x(\theta_1), s(\theta_1)) - \Psi(x, s) \\ & = (\rho - n) \log \frac{\mu(\theta_1)}{\mu} - \sum_{i=1}^n \log \frac{x(\theta_1)_i s(\theta_1)_i}{\mu(\theta_1)} + \sum_{i=1}^n \log \frac{x_i s_i}{\mu} \\ & \leq (\rho - n) \log \left[1 - \theta_1(1 - \gamma) + \frac{\|g(\theta_1)\|_{\infty}}{\mu} \right] + n \log \frac{1}{1 - \beta} \\ & \leq (\rho - n) \log [1 - \theta_1(1 - \gamma - n\theta_1 C^*)] + n \log \frac{1}{1 - \beta} \\ & \leq (\rho - n) \log \left[1 - \theta_1 \left(1 - \gamma - \frac{\beta\gamma}{2 - \beta} \right) \right] + n \log \frac{1}{1 - \beta} \\ & \leq -(\rho - n) \left(1 - \frac{2\gamma}{2 - \beta} \right) \theta_1 + n \log \frac{1}{1 - \beta} \\ & = -2n \log \frac{1}{1 - \beta}. \end{aligned}$$

So far, the Lemma is proved. □

Finally, we will finish the whole paper with the following main theorem.

Theorem 1. *The algorithm will terminate in $O(nL)$ iterations, where*

$$L = \log \frac{2^{\Psi^0/(\rho-n)}}{\varepsilon}, \quad \Psi^0 = \Psi(x^0, s^0).$$

Proof. If $(\rho - n) \log((x^k)^T s^k) \leq (\rho - n) \log \varepsilon$, then $(x^k)^T s^k \leq \varepsilon$. Applying the Proposition 1 we can have

$$(\rho - n) \log((x^k)^T s^k) \leq \Psi(x^k, s^k) \leq \Psi(x^0, s^0) - 2kn \log \frac{1}{1 - \beta}.$$

So the main task has to satisfy $\Psi(x^0, s^0) - 2kn \log \frac{1}{1 - \beta} \leq (\rho - n) \log \varepsilon$. Whence,

$$2kn \log \frac{1}{1 - \beta} \geq (\rho - n) \left[\frac{\Psi^0}{\rho - n} - \log \varepsilon \right] = \frac{3n^2(2 - \beta)C^*}{\beta\gamma \left(1 - \frac{2\gamma}{2 - \beta} \right)} \log \frac{1}{1 - \beta} \times \log \frac{2^{\Psi^0/(\rho-n)}}{\varepsilon}.$$

Therefore, we derive $k \geq \frac{3n(2-\beta)C^*}{\beta\gamma(1-\frac{2\gamma}{2-\beta})} \log \frac{2^{\Psi^0/(\rho-n)}}{\varepsilon} \geq n \log \frac{2^{\Psi^0/(\rho-n)}}{\varepsilon}$, which means that the algorithm need $O(nL)$ iterations at most to get a ε -solution. □

5 Conclusion

In this work, we present a new primal-dual interior-point algorithm for a class of convex programming under a scaled Lipschitz condition. The iterates generates

in a wide neighborhood of the central path, so it performs much better in implementation than its small-neighborhood counterparts. We demonstrated that it is a polynomial-time algorithm. However, the implementation of the algorithm remains to be the next work for the authors.

References

1. Hertog, D., Jarre, F., Roos, C., Terlaky, T.: A Sufficient Condition for Self-concordance, With Application to Some Classes of Structured Convex Programming problems. *Mathematical Programming* 69, 5–88 (1995)
2. Richalet, J.: Industrial Applications of Model Based Predictive Control. *Automatiza* 29, 1251–1274 (1993)
3. Zhang, M.W., Huang, C.C., Lai, Y.F.: An Interior Ellipsoid Method for Linearly Constrained Convex Programming. *Numerical Mathematics A Journal of Chinese Universities* 3, 215–221 (2000)
4. Karmarkar, N.K.: A New Polynomial-time Algorithm for Linear Programming. *Combinatorica* 4, 373–395 (1984)
5. Monteiro, R.D.C., Alder, I.: Interior Path Following Primal-dual Interior Point Algorithm Part II: Convex quadratic programming. *Mathematical Programming* 44, 43–66 (1989)
6. Monteiro, R.D.C., Alder, I.: An Extension of Karmarkar Type Algorithm to a Class of Convex Separable Programming Problems with Global Linear Rate of Convergence. *Mathematics of Operations Research* 15, 408–422 (1990)
7. Monteiro, R.D.C.: A Globally Convergent Primal-dual Interior Point Algorithm for Convex Problem. *Mathematical Programming* 64, 123–147 (1994)
8. Mizuno, S., Todd, M.J., Ye, Y.: On Adaptive-step Primal-dual Interior-point Algorithm for Linear Programming. *Mathematics of Operations Research* 18, 964–981 (1993)
9. Boyd, S., Crusius, C., Hansson, A.: Control Applications of Nonlinear Convex Programming. *Journal of Process Control* 8, 313–324 (1998)
10. Fang, S.C., Puthenpura, S.: *Linear Optimization and Extensions: Theory and Algorithms*[M]. Prentice Hall, New Jersey (1993)
11. Ye, Y.: *Interior Point Algorithms: Theory and Analysis*. John Wiley & Sons, Chichester (1995)
12. Zhao, Y.B., Han, J.Y.: Two Interior-point Methods for Nonlinear Complementarity Problems. *Journal of optimization theory and applications* 102, 659–679 (1999)
13. Nesterov, Y.E., Nemirovsky, A.S.: A General Approach to the Design of Optimal Methods for Smooth Convex Functions Minimization. *Ekonomika i Matem, Metody* 24, 509–517 (1988)
14. Nesterov, Y.E., Nemirovsky, A.S.: *Self-concordant Functions and Polynomial Time Methods in Convex Programming*. Central Economic & Mathematical Institute, USSR Academy of Science, Moscow, USSR (preprint, 1989)
15. Nesterov, Y.E.: *Interior Point Methods in Convex Programming: Theory and Applications*. Society for Industrial and Applied Mathematics, Philadelphia (1994)

An Improved Differential Evolution with Local Search for Constrained Layout Optimization of Satellite Module

Wang Chen, Yan-jun Shi, and Hong-fei Teng

School of Mechanical Engineering,
Dalian University of Technology, Dalian, P.R. China 116024
shiyj@dlut.edu.cn

Abstract. This paper proposes an improved differential evolution algorithm (called by DEG hereinafter) to tackle a kind of combinatorial optimization problem, i.e., constrained component layout optimization of satellite module. DEG combined local search with standard DE by employing the Gauss mutation operator from genetic algorithm, aiming to make random micro-perturbations to the location of components for obtaining a better layout scheme. The experimental results showed that the proposed DEG outperformed the standard DE and Genetic algorithm on layout design of simplified satellite module.

Keywords: Differential evolution; combinatorial optimization; genetic algorithm; layout design; satellite module.

1 Introduction

The Differential Evolution (DE) algorithm was a relative recent heuristic method introduced by Storn and Price[1] in mid-1990s. DE is a stochastic, population-based, and easy-used evolutionary algorithm and often was used to tackle the function optimization over continuous domains. For the discontinuous domain, e.g., 3D component layout[2], layout design of satellite module[3, 4] where the search space is discrete, non-linear and multi-modal, the traditional DE had be improved or modified to search such space. This paper studies layout design of a satellite module, considering how to place the given apparatuses and equipment (component) in the limited space of the satellite module. This placement must satisfy various behavioral constraints of the interior and exterior environment. The layout optimization of a satellite module is a key engineering problem in the schematic design of the satellite. For this problem, many researchers have proposed various computational intelligence methods, e.g., genetic algorithm [4], Hopfield neural network, Ant Colony Optimization (ACO)[5]. However, because of various requirements in the engineering, finding more alternative solutions was always useful for solving this problem. It is an open issue to find a better optimization method for obtaining a better solution of layout design. Until now, there is little work on this problem using differential evolution-based solvers.

Previous studies have shown that the various improved or modified versions of DE have been successfully applied to a variety of complex combinatorial optimization problems, e.g., job shop problems[6], aerodynamic optimization design[7], etc. However,

little work has been focused on improving differential evolution for constrained component layout optimization design. This study attempted to tackle this layout design of satellite module using the improving version of differential evolution with local search.

2 A Brief Introduction to Differential Evolution

Differential evolution (DE) is a population-based and directed search method [1]. Like other evolutionary algorithms (EAs), DE starts with an initial population vector, which is randomly generated when no preliminary experimental knowledge about the solution space is available. There are many schemes of the DE [8]. The particular version used in this study is the *DE/rand/1/exp* version, which appears to be the frequently used scheme[1], and is considered to be the basic version of DE.

Compared to other EAs, the DE needs less parameter. DE only controls N , F , and CR . It is important to note that, increasing either N or F to compute the mutation values also increases the diversity of possible movements, and promotes the exploration of the search space. Additionally, there control parameters also reduce the probability of obtaining the correct search direction. Therefore, we have to make initial experiment for obtaining the suitable algorithm parameters.

3 The Proposed Differential Evolution with Local Search

The previous studies have shown that DE has the characteristic of global parallel direct search for solving a variety of complex discontinuous and non-convex function. In the basic DE, crossover operator and mutation operator use the differential of population (see Eq.1 and Eq.2), and make searching optimal more quickly. However, to solve layout optimization problem, DE still needs special optimization operator for the combinatorial problem.

To find special optimization operator for DE, we firstly review the optimization operator from genetic algorithm in the literature. The optimization operators herein mean selection operator, crossover operator, and mutation operator. These operators from genetic algorithm for component layout design[5] included single-point crossover, two-point crossover, uniform crossover, Gauss mutation, etc. They were useful for obtaining a good layout topology by changing or swapping the location of component. These are the reasons why these operators were used in GA for solving layout optimization design problem. We borrowed the ideas from the aforementioned operators, and made an initial test to them. Then we found that Gauss mutation operator play an important role on location adjustment of component layout.

Gauss mutation operator has been widely employed in real-valued genetic algorithm and evolution strategy. In layout optimization design, this operator can adjust the position of layout objects by making perturbation to them. Therefore, we attempt to employ Gauss mutation to DE for obtaining better algorithm. Here, we proposed differential evolution with gauss mutation for local search (called by DEG hereinafter). Figure 1 shows the flowchart of DEG. For convenience, G , G_M denote the

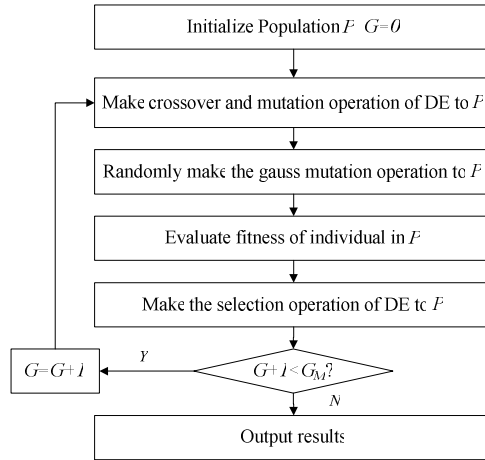


Fig. 1. Computational flowchart of differential evolution with gauss mutation

current generation and max generation of evolution process respectively. In each generation, DEG uses the crossover and mutation operation of DE to generate offspring, and then uses the gauss mutation to the offspring, and then uses the selection operation of DE to form a new population (the next generation). This evolution process will be terminated when $G = G_M$.

The key step of this DEG is the generation process of new individual $\bar{x}_{i,G+1}$ of population P_{G+1} . The process can be also described as follows.

(1) Mutation

For each target vector $x_{i,G}, i = 1, 2, \dots, N$, a mutant vector is generated according to

$$\bar{v}_{i,G} = \bar{x}_{i,G} + F[(\bar{x}_{best,G} - \bar{x}_{i,G}) + F[(\bar{x}_{r1,G} - \bar{x}_{r2,G})] \tag{1}$$

with random indexes $r1, r2 \in \{1, 2, \dots, N\}, r1 \neq r2 \neq i$, where $F \in [0, 2]$. The vector $\bar{x}_{best,G}$ denotes the best individual of generation G .

(2) Crossover

Generate the trial vector $\bar{x}'_{i,G} = [x'_{1i}, x'_{2i}, \dots, x'_{Di}]$ according to Eq.2.

(3) Gauss mutation

For each vector $\bar{x}'_{i,G}, i = 1, 2, \dots, N$, make gauss mutation operation to generate trial vector $\bar{x}''_{i,G}$ according to

$$x''_{ji,G} = x'_{ji,G} + N(0, \sigma_j), j = 1, 2, \dots, D, \tag{2}$$

where $N(0, \sigma_j)$ denotes gaussian random number, σ_j denotes the mutation step-size of j th element of vector $\bar{x}'_{i,G}$.

(4) Selection operator

The selection operator in DEG is similar to the DE. Using the current population P_G and the temporary population P''_{G+1} , DEG generate the population of the next generation P_{G+1} as

$$\bar{x}_{i,G+1} = \begin{cases} \bar{x}''_{i,G}, & \text{if } \Phi(\bar{x}''_{i,G}) < \Phi(\bar{x}_{i,G}) \\ \bar{x}_{i,G}, & \text{otherwise} \end{cases}, \tag{3}$$

where $\Phi(\bar{x})$ represents a fitness function.

To sum up, DEG obtain a new population P_{G+1} using the mutation, crossover, gauss mutation and selection operations described above, and make such iterations until meeting the ending conditions.

4 Constrained Layout Optimization Problem of Satellite Module

The optimal layout problem of a simplified satellite module can be described as follows[9]. A total number of N objects (component) must be located within a cylindrical satellite module. The module rotates around axis z at an angular velocity. The bearing plate in the middle of the module, a vertical plane of axis z , is used to fix all the objects. All the components are simplified as cuboids or cylinders in this study and regarded as rigid bodies with uniform mass distribution. The design objective here is to optimize the inertia performance of whole module, subjected to following constraints: (1) All the objects should be contained within the module, with no overlap among the objects and no clash between the module wall and each object; (2) Position error of the centroid of whole system should not exceed an allowable value, as small as possible; (3) Equilibrium error of the system should be permissible and, of course, the smaller the better.

This problem can also be regarded as a container loading problem where the container rotates and has a middle bearing plate to hold the objects. It belongs to three-dimensional packing problem with dynamical equilibrium constraints, where exists combination explosion in computing complexity.

Only cylinder objects will be located within the module in this work. In this work, R denotes the radius of the bearing plate and ω denotes the angular velocity of module. The i th object can be denoted by $X_i(O_i, r_i, m_i), i = 1, 2, \dots, N$, where $O_i = (x_i, y_i)$ is the center of the object X_i , and r_i, m_i is the radius and mass of X_i respectively. Then the mathematical model of above problem can be formulated as follows.

Find a layout scheme $X = \{X_i \mid i = 1, 2, \dots, N\}$, such that

$$\min f(X) = \min \left\{ \max \left\{ \sqrt{x_i^2 + y_i^2} + r_i \right\}, i = 1, 2, \dots, N, \right\} \tag{4}$$

subject to

$$h(X) = \sum_{i=1}^{N-1} \sum_{j=i+1}^N \Delta S_{ij} + \sum_{i=1}^N \Delta S'_i = 0, \tag{5}$$

$$g(X) = \sqrt{\left(\sum_{i=1}^N m_i x_i\right)^2 + \left(\sum_{i=1}^N m_i y_i\right)^2} \leq \delta_j, \tag{6}$$

where $f(X)$ denotes the maximum envelop radius of all the objects, $h(X)$ denotes the total overlap area among the objects and between the module wall and each object, ΔS_{ij} denotes the overlap area between X_i and X_j , $\Delta S'_i$ denotes the overlap area between the module wall and X_i , $g(X)$ is the distance between the centroid of the whole system and axis z , δ_j is the allowable error of the system should be permissible.

Here, the layout design of the satellite module was converted to an optimization problem without behavioral constraints using punishment coefficients method. The fitness function was used to evaluate the layout scheme and can be described as

$$F(X) = f(x) + w_1 \lambda_1 \max(0, g(X)) + w_2 \lambda_2 |h(X)|, \tag{7}$$

where $\lambda_i (i=1,2)$ are the normalization factors of each sub objective functions; $w_i (i=1,2)$ are the weight factors of each sub objective functions.

5 Experimental Study

To evaluate the proposed algorithms, we extended the case study introduced by Qian[3], where there were 14 cylinder objects that will be located on the 2 bearing surfaces in the simplified satellite’s module (see Fig. 2). Table 1 listed their dimensions and masses. The equilibrium error of the system should be smaller than a permissible value ($\delta_j = 3.4kg \cdot mm$). The radius of satellite module R was set to be $50mm$.

Table 1. The components’ geometry and dimension and mass

No	radius(mm)	mass(kg)	No	radius(mm)	mass(kg)
1	10.0	100.0	8	10.0	100.0
2	11.0	121.0	9	11.0	121.0
3	12.0	144.0	10	12.0	144.0
4	11.5	132.25	11	11.5	132.25
5	9.5	90.25	12	9.5	90.25
6	8.5	72.25	13	8.5	72.25
7	10.5	110.25	14	10.5	110.25

5.1 Experiment Setup

A fair time measure must be used to compare the different optimizers. As mentioned above, DEG uses gauss mutation operator for local search. If the number of generations

is used as a measure, it is unfair because DEG, DE and GA do the different amounts of fitness evaluations when optimizing the same problems. Considering that the number of function evaluations has a strong relationship with cost time, we used the number of function evaluations as a fair time measure in this study.

All experiments were run for 10^6 fitness function evaluations. And each of experiments was run 50 times with different random seeds. The reported results are the averages calculated from these 50 runs.

As mentioned above, the DE has three parameters: (1) the size of the population (N), (2) the crossover constant (CR), and (3) the scaling factor (F). Ref [1] stated that the size of the population increases as the dimension of function increases. As a result, they were set to the following values: $N = 100, CR = 0.8, F = 0.5$. These parameters follow the suggestion in other literature where they have been found empirically to obtain good performance[1], [8].

The GA here used real-coded representation, tournament selection, arithmetic crossover and Gaussian mutation[10]. The parameters of GA here were used the same setting as Ref.[3], which is effective for layout optimization problem. Besides, the population N , crossover probability P_c and mutation probability P_m of GA was respectively set to 50, 0.95 and 0.3.

5.3 Experimental Results and Discussion

To test each algorithm’s performance, three algorithms independently run 50 times respectively. The experimental results are shown in Table 2, Table 3 and Table 4. Table 2 shows the statistic results, including the fitness values and the computing time cost of each algorithm. And Table 3 lists the statistic results of performance, including the maximum enveloped radius, overlap area and static non-equilibrium. Then Table 4 lists the best layout schemes. Additionally, Fig. 3 shows the corresponding two-dimensional layout diagrams, including the upper and lower bearing plate.

Table 2 and Table 3 showed that DEG outperformed DE and GA on all of the layout performance, where the average fitness value of DEG increased by 2.739% and 58.72% respectively than DE and GA. The best fitness value of DEG is approximately equal to the one acquired from that of DE, and increased by 46.28% than that

Table 2. the comparative results of DEG, DE and GA

	Fitness values				Computing time /s			
	Mean	SD	Best	Worst	Mean	SD	Best	Worst
DEG	7.1729e+0	4.07e+	6.7000e+0	7.5360e+0	3.4938e+0	2.35e-01	3.4592e+0	3.54e+0
	1	0	1	1	1		1	1
DE	7.3694e+0	2.83e+	6.7569e+0	8.7400e+0	3.2060e+0	2.70e-01	3.1507e+0	3.26e+0
	1	0	1	1	1		1	1
GA	1.1385e+0	5.85e+	9.8010e+0	1.1944e+0	3.4905e+0	2.31e-01	3.4567e+0	3.54e+0
	2	0	1	2	1		1	1

Table 3. the comparative performance results of DEG, DE and GA

	maximum envelop radius (mm)				Overlap area(mm ²)		static non-equilibrium(kg · mm)			
	Mean	SD	Best	Worst	Mean	SD	Mean	SD	Best	Worst
DEG	35.5431	4.21	32.4777	38.9033	0.0000	0.00	0.0000	0.00	0.0000	0.0000
DE	35.6602	3.98	32.7194	39.6072	0.0000	0.00	0.0000	0.00	0.0000	0.0000
GA	46.6098	3.54	44.4474	53.0508	14.431	5.42	2.1060	1.87	1.0004	3.9346

Table 4. the comparative best layout results of DEG, DE and GA

No.	DEG		DE		GA	
	$x(mm)$	$y(mm)$	$x(mm)$	$y(mm)$	$x(mm)$	$y(mm)$
1	0.000	24.849	0.000	21.343	0.000	-30.185
2	-6.284	-22.458	20.765	12.382	-22.789	-17.343
3	21.115	8.732	-19.633	8.359	27.410	6.900
4	15.514	-16.887	5.729	-23.064	-19.879	8.185
5	-18.001	17.847	-22.384	-14.490	-4.697	29.962
6	-0.359	-4.406	21.435	-7.961	15.757	-11.368
7	-24.329	-0.791	2.525	-1.261	5.375	9.635
8	0.000	1.023	0.000	-9.501	0.000	-14.785
9	-16.793	13.743	20.467	1.543	-35.112	-12.537
10	-2.718	-20.540	-7.840	12.632	5.775	33.309
11	21.808	4.057	18.058	-24.044	-7.612	-36.860
12	-22.306	-6.323	14.287	29.220	39.358	-5.010
13	17.108	-16.342	-29.424	16.536	-26.254	10.569
14	4.059	21.842	-26.335	-17.198	27.666	28.270

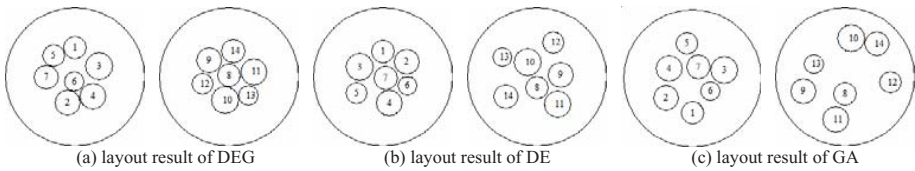


Fig. 3. The best layout result of DEG, DE and GA

of GA. From Table 3, we can see that DEG and DE were able to search feasible layout scheme, that is, overlap area is 0 and static non-equilibrium smaller than δ_j , through 10^6 times or less evaluation of fitness function. But GA did not do that. Besides, the average maximum envelop radius of DEG decreased by 0.328% and 23.74% respectively than DE and GA. Therefore, the performance of DEG was superior to DE and GA in this experiment. However, Table 2 also showed that the efficiency of DEG is lower than DE and GA. Among three algorithms, the average calculating time cost of DE is shortest and decreased by 8.237% than DEG, and GA is almost the same as DEG.

From the above results, we found that DEG was superior to DE and GA on performance. The reason may be that the Gauss operator of DEG can constantly micro-adjust the location of component for obtaining a better placement. This operator help DEG obtain a more satisfied layout solution. We have to note that DEG spent more time than DE and GA. In engineering practice, we believe that the more cost time may be acceptable.

6 Conclusion

We improve the basic DE with local search for constrained component layout optimization of satellite module, aiming at obtaining a better solution-scheme. The problem in this study is very difficult to solve in polynomial time^[5]. There was still no perfect solution-scheme in previous studies. We proposed the DEG for a helpful exploration

of this problem. The experimental results indicates that the performance of DEG over DE and GA with this kind of problem. Furthermore, this work would be extended to apply in layout design of tank, submarine, etc.

Acknowledgements

This work was supported by National Natural Science Foundation of P R China (Grant No. 50575031), National High-tech R&D program of P R China (Grant No. 2006AA04Z109) and National Defense Basic Scientific Research Project of P R China (Grant No. B0920060901).

References

1. Storn, R., Price, K.: Differential Evolution—A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization* 11(4), 341–359 (1997)
2. Aladahalli, C., Cagan, J., Shimada, K.: Objective Function Effect Based Pattern Search: Theoretical Framework Inspired by 3D Component Layout. *Journal of Mechanical Design* 129(3), 243–254 (2007)
3. Qian, Z.Q., Teng, H.F., Xiong, D.L.: Human-Computer Cooperation Genetic Algorithm and its Application to Layout Design. In: *Proceeding of the 4th Asia-Pacific Conference on Simulated Evolution and Learning*, Singapore, pp. 299–302 (2002)
4. Grignon, P.M., Fadel, G.M.: A GA Based Configuration Design Optimization Method. *Journal of Mechanical Design* 126, 6–15 (2004)
5. Sun, Z.G., Teng, H.F.: Optimal Layout Design of a Satellite Module. *Engineering Optimization* 35(5), 513–529 (2003)
6. Qian, B., Wang, L., Huang, D.X.: Scheduling Multi-Objective Job Shops Using A Memetic Algorithm Based on Differential Evolution. *International Journal of Advanced Manufacturing Technology* 35(9-10), 1014–1027 (2008)
7. Song, L.M., Li, J., Feng, Z.P.: Aerodynamic Optimization Design of Low Aspect Ratio Transonic Turbine Stage. *Chinese Mechanical Engineering Society* 19(4), 500–504 (2006)
8. Storn, R.: System Design by Constraint Adaptation and Differential Evolution. *IEEE Transactions on Evolutionary Computation* 3(1), 22–34 (1999)
9. Teng, H.F., Sun, S.L., Liu, D.Q.: Layout Optimization for the Objects Located within A Rotating Vessel – A Three-Dimensional Packing Problem with Behavioral Constraint. *Computers & Operations Research* 28, 521–535 (2001)
10. Baeck, T., Fogel, D.B., Michalewicz, Z.: *Handbook of Evolutionary Computation*. Institute of Physics Publishing (1997)

Cloud Theory Based Simulated Annealing Algorithm for Multiple Observers Sitting on Terrain Problem

Pin Lv^{1,2}, Lin Yuan^{1,2}, and Jinfang Zhang¹

¹ National Key Laboratory of Integrated Information System Technology, Institute of Software, Chinese Academy of Sciences, Beijing 100190

² Graduate University, Chinese Academy of Sciences, Beijing 100049

Abstract. The problem of Multiple Observers Sitting on Terrain (MOST) is an important part in visibility-based terrain reasoning and many applications can be classified as this problem. Recent developments in this field concentrate on using heuristic algorithm, such as Simulated Annealing algorithm (SA), but it is still difficult because of unacceptable computing time and low solving precision. In this paper, a Cloud theory based Simulated Annealing algorithm (CSA) is introduced involving two innovations. The first is state changing by using X cloud generator which can make the position selection of observer controllable. The second is temperature annealing by using Y cloud generator which can produce approximatively continuous annealing temperature and fit the physical annealing process in nature much better. Theoretical analysis proves that CSA is convergent for MOST problem. Application experiments show that, using CSA for MOST problem, the average time cost decreases by 40%~60% and the average solution accuracy improves by 10% as compared with using SA.

1 Introduction

The Multiple Observers Sitting on Terrain (MOST) problem is a combinatorial optimization problem which consists of finding the fewest possible observers to make the united viewshed of those observers cover a certain ratio area, given the kind of the observer (person, radar, etc.) and the attributes of the observer (height, the radius of ROI, etc.). It is an important part in Visibility-Based Terrain Reasoning (VBTR). This problem was identified as far back as 1982 by Nagy [1]. And many applications can be classified as the MOST problem, such as locating a telecommunication base station [2], protecting endangered species [3], locating wind turbines [4] and so on.

As the problem scale enlarging, it becomes difficult to solve because of the unacceptable computing time and low solving precision. Therefore, it is necessary to find new method to face this challenge. Recent developments in this field focus on involving spatial optimization techniques, such as heuristic algorithm. For example, [5] used the swap algorithm in their toolkit for solving MOST problem. [6] compared the results of using different heuristic algorithm individually for

MOST problem and claimed that the Simulated Annealing algorithm (SA) is the best one among all tested algorithms when considering the balance between time cost and solution accuracy.

Based on the fundamental concepts of cloud theory, a Cloud theory based Simulated Algorithm (CSA) is explained in detail in Section 2. It contains two main contributions, the first is state changing by using X cloud generator which can make the observers position changing be controllable and improve the searching ability, and the second is temperature annealing by using Y cloud generator which can produce nearly continuous annealing temperature in each searching step and fit the physical rules better. Section 3 proves that CSA for MOST problem is convergent. Experiment results are shown in section 4. They show that the CSA can enhance the searching for veracity and reduce the time cost.

2 Cloud Theory Based Simulated Annealing Algorithm for MOST

2.1 Basic Concepts of Cloud Theory

Consider a quantitative domain U which is represented by accurate numerical value and can have an arbitrary number of dimensions and C is a qualitative concept under U . If quantitative value $x \in U$ and x is a stochastic realization of C in qualitative concept. The certainty degree of x to C , $u(x) \in [0, 1]$, is a random number with stable tendency u that satisfies that when $U \rightarrow [0, 1]$ and $x \in U$, then $x \rightarrow u(x)$. Then, the distribution of x on domain U comes to be known as a cloud model, and is called cloud for short [7].

2.2 State Changing by Using X Condition Cloud Generator

The state changing of traditional SA for MOST is un-controllable. If generate a new observer's position s_j from its current position s_i , the common method is

$$s_j = s_i + rand(-1, 1) \cdot Max \tag{1}$$

Where $rand(a, b)$ function generates random numbers which uniformly distribute between a and b , and Max is the maximal distance between every two positions.

Using X condition cloud generator and taking a certain value as a reference, it can randomly generates a group of new values which distribute around the given reference like "cloud" and the density of this cloud can be controlled by He . The new state changing process is described as follows.

$$He = t_k, \quad x = 1.0 - t_k \tag{2}$$

Where t_k is the temperature of the No. k annealing step of SA.

$$En' = \frac{1}{\sqrt{2}}[En + \sqrt{He} \cdot rand(0, 1)] \tag{3}$$

$$u_i = \exp \left[-\frac{1}{2} \cdot \left(\frac{x - Ex}{En'} \right)^2 \right] \tag{4}$$

$$s_j = s_i + u_i \cdot Max \tag{5}$$

Because the annealing temperature $t_k \geq 0$, then let $Ex = 0$ in the CSA algorithm, and according to “3En” rule, then set $En = 0.33$, which can meet X cloud generator’s requirement for the range of discussion domain u .

Figure 1 is the comparison of unified searching range during each annealing step between CSA and SA for MOST when states change 100 times in each annealing step. From this figure, it is clear to see, during the first three annealing steps, the searching range of CSA is wide which is helpful to let the observer’s position locate as vast as possible so as to preserve diversity of the searched individual. As the annealing times increasing and the annealing temperature falling, the searching range of CSA becomes narrow which is useful for fixing the observer’s optimal position quickly.

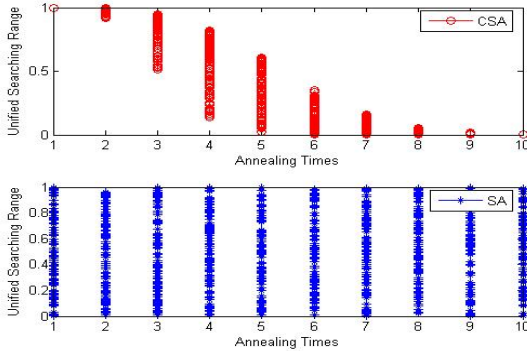


Fig. 1. Comparison unified searching range for two algorithms, $\lambda=0.9$

2.3 Temperature Annealing by Using Y Condition Cloud Generator

The temperature annealing function of SA is exponential in general [8]. Suppose the initial temperature is t_0 , the annealing index is λ , then the annealing temperature in the No.k step is

$$t_k = t_0 \cdot \lambda^k \quad k = 1, 2, 3, \dots, 0 < \lambda < 1 \tag{6}$$

Using Y condition cloud generator and taking a certain value as a reference, it can also randomly generate a group of new values which distribute around the given reference like “cloud”. The new temperature annealing process is described as follows.

$$He = t_k, \quad En = t_k, \quad u_0 = 1.0 - t_k \tag{7}$$

$$En' = \frac{1}{3}[En + He \cdot rand(0, 1)] \tag{8}$$

$$t'_k = En' \sqrt{-2 \ln u_0} \tag{9}$$

Where t'_k is the changeable annealing temperature of each state changing. Because He determines the dispersive degree of cloud drop, He in the CSA is direct proportion to the reference temperature t_k .

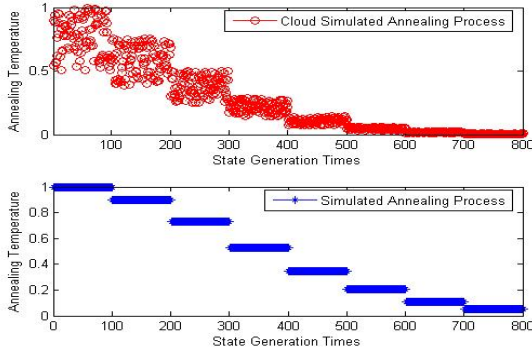


Fig. 2. Comparison of annealing temperature for two algorithms, $\lambda = 0.9$

Figure 2 is the comparison of annealing temperature with each state between CSA and SA when states change 800 times. From this figure, it is clear to see that CSA can preserve the characteristic of annealing temperature’s gradual descending as well as SA. At the same time, the changeable bound of annealing temperature of CSA is far wider than that of SA and approximately continuous, then CSA overcomes the disadvantage of SA whose temperature is discrete and simplex during each annealing step, so the annealing process of CSA is coincident with the physical annealing process better.

2.4 Algorithm Flow

Define the annealing temperature as t_k , the new annealing temperature as t'_k and the positions of observer set as s_k . Set the initial temperature be $t_0 = 1.0$, the initial positions of observer set be s_0 and quit temperature be t_q . Let $C(s_k)$ represent the united-viewshed coverage ratio of the observer set. Then the CSA algorithm for MOST problem is as follows.

3 Analysis of CSA’s Convergence for MOST Problem

Theorem 1. *The CSA is corresponding to a Markov chain.*

Proof. The Markovian property of CSA is related with its state generator function illustrated in section 2.2:

$$V = \frac{1}{2}[E_n + \sqrt{T} \cdot rand(0, 1)] \tag{10}$$

$$X_{n+1} = X_n + Max \cdot \exp \left[-\frac{(1 - T)^2}{2V^2} \right] \tag{11}$$

Algorithm 1. CSA for MOST

```

1: procedure CSA
2:    $t_k = t_0, s_k = s_0, k = 0;$ 
3:   for  $t_k > t_q$  do
4:     for Metropolis Rule is not satisfied do
5:       Generate  $t'_k$  using eq (7) to (9);
6:       Generate  $s_j$  from  $s_i$  using eq (2) to (5);
7:       Compute  $C(s_k)$ ;
8:       if  $\min\{1, \exp[-\frac{C(s_i)-C(s_j)}{t'_k}]\}$   $> \text{rand}(0, 1)$  then
9:          $s_k = s_j;$ 
10:      else
11:         $s_k = s_i;$ 
12:      end if
13:    end for
14:     $k = k + 1,$  Generate  $t_k$  using eq (6);
15:  end for
16:  return  $s_k;$ 
17: end procedure

```

According to the state generating method of CSA, the generation of next state X_{n+1} is only correlative with current state X_n . Therefore, this state transferring has Markovian property.

The transferring probability for next annealing step under the temperature T of this Markov chain is:

$$p_{i,j}(T) = \begin{cases} g_{i,j}(T)a_{i,j}(T), & j \in D_i \text{ and } j \neq i \\ 1 - \sum_{k \in D_i} p_{i,k}(T), & j = i \\ 0, & \text{others} \end{cases} \tag{12}$$

$g_{i,j}$ is the probability of the state transferring from i to j under the temperature T , and $a_{i,j}(T)$ is the acceptance function under this temperature, D_i is the neighbor domain of state i .

According to the analysis of CSA in the section 2.3, the acceptance function $a_{i,j}(T)$ can be presented as:

When $C(j) \leq C(i)$, then $a_{i,j}(T) = 1$; When $C(j) > C(i)$, then

$$a_{i,j}(T) = \int f_\tau(\xi, T) e^{-\frac{C(j)-C(i)}{\xi}} d\xi \tag{13}$$

Theorem 2. *The state space of CSA's Markov chain is limited.*

Proof. For MOST problem, if s observers are located on a terrain which has sample $m \times n$ points, then all the possible choices of observer's position combination composes the state space for this problem. Therefore, the state space for MOST is $C_{m \times n}^s$, and it shows that the CSA is corresponding to a limited Markov chains.

Lemma 1. *All states of the Markov chain whose state is limited and unreduced has natural returning.*

Theorem 3. *This state limited Markov chain can not be reduced.*

Proof. According to the neighbor selection method and state generation function, strong connectivity of the feasible domain guarantees the unreducibility. $\forall i, j \in \Omega, \exists s_0, s_1, \dots, s_n \in \Omega, s_0 = i, s_n = j$, and every state is not same as each other, the property of strong connectivity ensures that $g_{s_k, s_{k+1}} > 0, k = 0, 1, \dots, n - 1$. Moreover, because $\forall i, j \in \Omega, a_{i,j}(T) > 0$

$$p_{i,j}^n(T) \geq p_{s_0, s_1}(T)p_{s_1, s_2}(T) \cdots p_{s_{n-1}, s_n}(T) > 0 \tag{14}$$

the arbitrary state of this Markov chain is connectivity each other. Therefore, this Markov chain can not be reduced.

Theorem 4. *This state limited Markov chain is non-periodic.*

Proof. Because this state limited Markov chain can not be reduced, then all states of this chain is equivalent and have same period.

Consider the state acceptance function $a_{i,j}(T)$. When $C(j) \leq C(i)$, then $a_{i,j}(T) = 1$. When $C(j) > C(i)$, then

$$0 < a_{i,j}(T) < \int |f_\tau(\xi, T)| d\xi = 1 \tag{15}$$

If $\exists i \in \Omega, j \in D_i$, which are satisfied with $0 < a_{i,j}(T) < 1$, consider the transferring probability of i :

$$p_{i,i}(T) = 1 - \sum_{k \in \Omega, k \neq i} p_{i,k}(T) > 1 - \sum_{k \in \Omega, k \neq i, k \neq j} g_{i,k} - g_{i,j} = 0 \tag{16}$$

Thus the state i is non-periodic. And also because all states have same period, then this Markov chain is non-periodic.

Lemma 2. *The state limited, un-reduced and non-periodic Markov chain has unique smooth distribution.*

Lemma 3. *The sufficient and necessary condition of natural returning for a chain which is non-periodic and un-reduced is that it has smooth distribution, and this distribution is its ultimate distribution.*

Theorem 5. *The Markov chain of CSA for MOST problem is convergent.*

Proof. This is the direct conclusion from theorem 1~4 and lemma 1~3.

Table 1. Statistical characteristic of sample terrain

Sample	Min	Max	Diff	Mean	SD
1	693.1	754.2	61.1	712.2	84.28
2	939.3	2531.5	1592.2	1731.2	250.34
3	250.0	461.3	211.3	364.4	1272.6
4	2153.5	2570.1	416.6	2372.7	2030.2
5	930.4	2481.7	1551.3	2023.6	3677.1

4 Using CSA for MOST Problem

We solve the MOST problem using SA and CSA respectively for five representative terrains (described in Table 1) in two experiments. In experiment 1, there are 4 blocks and 2 observers per block. And in experiment 2, there are 4 blocks and 4 observers per block. The observer's ROI are 256 sample points, annealing index λ is 0.9 and observer's height is 1.6 meters in both experiments. Each experiment is repeated 10 times for each terrain. The two algorithms stop when the temperature descends to 10% of the original temperature stage. All the experiments are done by using a PC which has 2.4 GHz Pentium CPU and 1 Gbytes RAM to get the comparison of time cost (TC, unit is second) and united-viewshed coverage (UVC, unit is %) between using SA and CSA. The experiment results are presented below.

From table 2, it is clear to see that the average time cost of the solution based on the CSA decreases by 40%~60% and the accuracy improves by 10% as compared with the one based on the SA. It illustrates that the gain of using CSA is much more than using SA if the complexity of the MOST problem increases.

Table 2. Comparison result of united-viewshed coverage ratio and the time cost

Sample	Experiment 1		Experiment 2		Performace
	SA	CSA	SA	CSA	
1	706.4	503.4	5633	2484	TC(sec)
	25.67	27.91	35.69	40.12	UVC(%)
2	719.2	522.4	6333	1742	TC(sec)
	31.52	33.79	45.14	49.36	UVC(%)
3	650.0	463.8	3529	2638	TC(sec)
	25.43	29.84	36.48	42.16	UVC(%)
4	595.1	398.4	1751	867.8	TC(sec)
	19.39	22.71	28.26	33.59	UVC(%)
5	722.2	512.3	5505	1769	TC(sec)
	24.26	25.91	36.21	39.2	UVC(%)

5 Conclusions

In VBTR, using a heuristic algorithm is an efficient method to solve the MOST problem. However, if we use a general heuristic algorithm without any modification, it usually cannot get the best effect of the balance between efficiency and precision. In this paper, according to the cloud drop's randomness and stable tendency in cloud theory, a Cloud theory based Simulated Annealing algorithm (CSA) is developed. CSA not only realizes the controllable changing of observer's position by using X cloud generator, but also fits the physical annealing process in nature much better by using Y cloud generator which can produce nearly continuous annealing temperature. It has been proved that the CSA is convergent for MOST problem theoretically. Moreover, the result of the application of using CSA for MOST problem declares that the new algorithm's usefulness and effectiveness adequately and it can be used in other combinational optimization applications of terrain reasoning.

References

1. De Floriani, L., Falcidieno, B., Pienovi, C., et al.: A Visibility Based Model for Terrain Features. In: Proceedings of Second international symposium on Spatial Data Handling, pp. 235–250 (1986)
2. Anderson, H., McGeehan, J.: Optimizing Microcell Base Station Locations Using Simulated Annealing Techniques. In: Proceedings of IEEE VTC 1994, vol. 2, pp. 858–862 (1994)
3. Camp, R., Sinton, D., Knight, R.L.: Viewsheds: A Complementary Management Approach to Buffer Zones. *Wildlife Society Bulletin* 25, 612–615 (1997)
4. Kinder, D., Sparkes, A., Dorey, M.: GIS and Wind Farm Planning. Springer, Heidelberg (1999)
5. Franklin, W.R., Metin, I., Zhongyi, X.: Smugglers and Border Guards: the Geostar Project at rpi. In: Proceedings of the 15th annual ACM international symposium on Advances in geographic information systems (2007)
6. Kim, Y., Rana, S., Wise, S.: Exploring Multiple Viewshed Analysis Using Terrain Features and Optimisation Techniques. *Computers and Geosciences* 30, 1019–1032 (2004)
7. Fang, W., Yanpeng, L., Xiang, L.: A New Performance Evaluation Method for Automatic Target Recognition Based on Forward Cloud. In: Proceedings of Asia Simulation Conference 2007, pp. 337–345 (2007)
8. Laarhoven, V., Aarts, E.: Simulated Annealing: Theory and Applications. Kluwer Academic, Norwell (1987)

Minimum Vertex Ranking Spanning Tree Problem on Some Classes of Graphs

Ruei-Yuan Chang, Guanling Lee, and Sheng-Lung Peng*

Department of Computer Science and Information Engineering
National Dong Hwa University
Hualien 974, Taiwan
slpeng@mail.ndhu.edu.tw

Abstract. A vertex ranking of a graph G is a labeling of the vertices of G with positive integers such that every path between two vertices with the same label i contains a vertex with label $j > i$. A vertex ranking is minimum if the largest label used in it is the smallest among all possible vertex rankings of G . The minimum vertex ranking spanning tree problem on G is to find a spanning tree T of G such that the minimum vertex ranking of T is minimum among all possible spanning trees of G . In this paper, we show that the minimum vertex ranking spanning tree problem on interval graphs, split graphs, and cographs can be solved in linear time. It improves a previous result that runs in $O(n^3)$ time on interval graphs where n is the number of vertices in the input graph.

1 Introduction

Let $G = (V, E)$ be a finite, simple, and undirected graph. Let $n = |V|$ and $m = |E|$. Let $N(v) = \{u \mid (u, v) \in E\}$. A *clique* (respectively, *independent set*) is a vertex subset $W \subseteq V$ such that any two vertices in W are adjacent (respectively, non-adjacent). Let \overline{G} be the complement of G , i.e., $\overline{G} = (V, \overline{E} = \{(u, v) \mid (u, v) \notin E\})$. The *diameter* of G is the longest shortest path among all possible shortest paths in G .

A *vertex ranking* of a graph G is a function $\gamma : V \rightarrow \mathbb{N}$ such that every u, v path in G with $\gamma(u) = \gamma(v)$ contains at least one vertex w in the path with $\gamma(w) > \gamma(u)$. In a vertex ranking γ , $\gamma(v)$ is called the *rank* of v . A vertex ranking γ is *minimum* if the largest rank used in γ is the minimum among all possible vertex rankings of G . We denote the largest rank used in the minimum vertex ranking of G by $\text{rank}(G)$. The *minimum vertex ranking problem* on G is to find a minimum vertex ranking of G . This problem has applications to, e.g., communication network design, planning efficient assembly of products in manufacturing systems [3, 5, 6, 11], and VLSI layout design [2, 7].

The *minimum vertex ranking spanning tree* (MVRST for short) problem on G is to find a spanning tree of G such that the vertex ranking of the spanning tree is minimum among all possible spanning trees of G [9]. Miyata *et al.* proved that the decision version of MVRST problem is NP-complete on general graphs

* Corresponding author.

[8]. However, it can be solved in $O(n^3)$, $O(n^5)$, and $O(n^5 \log^4 n)$ time on interval graphs [9], outerplanar graphs [10], and series-parallel graphs [11], respectively.

In this paper, we propose linear-time algorithms for the MVRST problem on interval graphs, split graphs, and cographs. Our result on interval graphs improves the result of [9] from $O(n^3)$ to linear time.

2 Interval Graphs

A graph G is called an *interval graph* if it has an intersection model F consisting of intervals on a straight line such that any vertex of G corresponds to an interval in F , and two vertices are adjacent if their corresponding intervals overlap.

Masuyama and Nakayama showed that the MVRST problem on interval graphs can be solved in $O(n^3)$ time [9]. The basic idea proposed in [9] is as follows. At first, a diameter P of G is determined. Note that P must contain the leftmost vertex v_{min} and the rightmost vertex v_{max} of G , where v_{min} (respectively, v_{max}) is the vertex with the least right endpoint (respectively, the largest left endpoint) [9]. In the following, we only consider such a diameter.

Next, a spanning tree T with minimum vertex ranking is computed by linking each $v \in V \setminus P$ to a vertex of P or to a vertex which is already linked to P , or adding v into P . In fact, T is determined by using a dynamic programming technique on P in [9]. To improve the time complexity, we try to construct the MVRST directly without considering some cases. By the lemmas proved in [9], we have the following lemma.

Lemma 1. *Let P be a diameter of G and T be an MVRST of G . Then $rank(P) \leq rank(T) \leq rank(P) + 1$.*

In our approach, during the construction of a spanning tree T of G , we try to extend P to P' such that $rank(P) = rank(P') = rank(T)$ until it is impossible to extend P . We then conclude that $rank(T) = rank(P) + 1$. By Lemma 1, it is optimal. The key point is that the minimum vertex ranking of a path is not unique. Thus we try to make the ranking unique. Lemma 2 shows that the minimum vertex ranking is unique for some special paths and it can be proved by induction.

Lemma 2. *Let P be a path with $|P| = 2^r - 1$ for some $r \in \mathbb{N}$. Then, the minimum vertex ranking of P is unique and $rank(P) = r$.*

By Lemmas 1 and 2, we have Lemma 3.

Lemma 3. *Let P_1 and P_2 be two paths such that $2^{\lfloor \log_2 |P_1| \rfloor} - 1 < |P_1| \leq 2^{\lfloor \log_2 |P_1| \rfloor + 1} - 1 = |P_2|$. Then, $rank(P_1) = rank(P_2)$. Moreover, if P_3 is a path with $|P_3| > |P_2|$, then $rank(P_3) > rank(P_2)$.*

If $|P| < 2^r - 1$ for some $r \in \mathbb{N}$, we try to extend P as long as possible by Lemma 3. If we can add all the vertices in $V \setminus P$ before extending P within $2^{\lfloor \log_2 |P| \rfloor + 1} - 1$ vertices, we obtain an MVRST whose rank is equal to $rank(P)$.

Otherwise, a spanning tree T can be obtained by the following operation. Let $P = [u_1, u_2, \dots, u_d]$ and γ' be an optimal ranking on $P \setminus \{u_1, u_d\}$. The *levelup* operation on P is a ranking γ on P by letting $\gamma(u_1) = \gamma(u_d) = 1$ and $\gamma(u_i) = \gamma'(u_i) + 1$ for $1 < i < d$. Then we can directly link every vertex $v \in V \setminus P$ to a vertex in $\{u_2, \dots, u_{d-1}\}$ by letting $\gamma(v) = 1$. It lets us obtain a vertex ranking spanning tree of G . In this paper, we call the number $2^{\lfloor \log_2 |P| \rfloor + 1} - 1$ the *corresponding minimum vertex ranking bound* (CMVRB for short) of the path P . The following lemma shows that some MVRST of G can be determined by its diameter P .

Lemma 4. *Let P be a diameter of G . If $|P|$ is either $2^{\lfloor \log_2 |P| \rfloor}$ or $2^{\lfloor \log_2 |P| \rfloor + 1}$, then the rank of the MVRST of G is $\lfloor \log_2 |P| \rfloor + 1$.*

Let P be a diameter of the interval graph $G = (V, E)$. In [9], vertices in $V \setminus P$ can be partitioned into three subsets, namely, V_1 , V'_1 , and V_2 as follows.

- V_2 contains the vertices which are adjacent to more than one vertex of P .
- V_1 contains the vertices that are exactly adjacent to one vertex of P and not adjacent to any vertex of V_2 .
- V'_1 contains the vertices that are exactly adjacent to one vertex u of P and adjacent to at least one vertex of V_2 .

For our algorithm, we partition V'_1 into three subsets, namely, V'_{1-1} , V'_{1-2-1} , and V'_{1-2-2} as follows.

- $V'_{1-1} = \{v \in V'_1 \mid \exists v' \in V_1 \text{ s.t. } N(v') \cap N(v) \cap P = \{u\}\}$.
- $V'_{1-2-1} = \{v \in V'_1 \mid v \notin V'_{1-1} \text{ s.t. } v \text{ can link to both of } N(u) \cap P \text{ via some vertices in } V_2\}$.
- $V'_{1-2-2} = \{v \in V'_1 \mid v \notin V'_{1-1} \text{ s.t. } v \text{ can link to only one of } N(u) \cap P \text{ via some vertices in } V_2\}$.

Since vertices in P that are adjacent to the same vertex of V_2 must be consecutive, vertices in V'_{1-2-1} can connect to a vertex in P whose rank is more than 2 via a vertex in V_2 . Thus we have the following lemmas.

Lemma 5. *Connecting V_2 to P does not change the ranking of P .*

Lemma 6. *Connecting V'_{1-2-1} to P does not change the ranking of P .*

By Lemmas 5 and 6, we can process the vertices in V_2 and V'_{1-2-1} after solving all the other cases. Consider the vertices in V_1 . Since they have only one choice to connect to P , we connect vertices of V_1 to P directly. In fact, vertices in V'_{1-1} can be treated as vertices in V_1 . Thus we have the following lemma.

Lemma 7. *Connecting V'_{1-1} to P does not change the ranking of $P \cup V_1$.*

Thus, all we have to do is to connect vertices of V'_{1-2-2} to P such that the ranking of P is as unchanged as possible. Since the vertices in P which are connected to

V_1 and V'_{1-1} must be ranked more than 1 in some minimum vertex ranking, we mark these vertices *black* to denote these vertices to be ranked more than 1. We also mark the second and the second to the last vertex of P black since the first and the last vertex of P can be treated as the vertices in V_1 or V'_{1-1} .

For an arbitrary minimum vertex ranking of P , if two adjacent vertices or an end vertex are ranked more than 1, there can be an ignored hidden vertex which is ranked as 1. In this paper, we call such an ignored hidden vertex a *division*. Clearly, if two adjacent vertices in P are marked black, then there exists a division between these two vertices since both of them must be ranked more than 1. Let D_s be the set of all divisions in P .

These black vertices partition P into P_1, P_2, \dots, P_s with $|P_i| > 0$ for $1 \leq i \leq s$. For the path P^* with $CMVRB$ vertices, since all the black vertices in P must be ranked more than 1, we have to keep the black vertices to be located in the even position in P^* . Note that we also count position of each division. So we can see that there exists a division in each subpath with even number of vertices by the properties of minimum vertex ranking. We denote $\mathbb{P}^{even} = \{P_i | 1 \leq i \leq s \text{ and } |P_i| \text{ is even}\}$. Then we can count the number of divisions in even subpaths by $|\mathbb{P}^{even}|$.

Let *free vertex number* (FVN for short) be the number of vertices that can be added into P such that the number of vertices in the resulting path is $CMVRB$. Then we can compute FVN by using the $CMVRB$ to subtract $|D_s|$, $|\mathbb{P}^{even}|$, and $|P|$. Now, we are going to connect the vertices in V'_{1-2-2} to P .

There are three ways to connect $v \in V'_{1-2-2}$ to P . Let $u \in N(v) \cap P$. The first way is to connect v to u directly when $\gamma(u) > 1$. The second way is to connect v to $u' \in P$ via some vertex in V_2 when $\gamma(u) = 1$ and $\gamma(u') > 2$. These two ways would not decrease FVN since they do not change the minimum vertex ranking and the order of P . When both two ways cannot work, then we can only do the last way. The last way is that we first choose a vertex w from $N(v) \cap V_2$ which connects to most vertices in V'_{1-2-2} . Then we add w into P and the vertices in $N(w) \cap V'_{1-2-2}$ can be put into V_2 . However, the last operation makes $|P|$ increase 1 and FVN decrease 1. If $FVN = 0$, $V'_{1-2-2} \neq \emptyset$, and need to do the last connecting way, this will increase the *rank*(P) for adding the remaining vertices to the spanning tree. That is, we have to do *levelup* on P . Figure 1(a), (b), and (c) show the three connecting ways respectively.

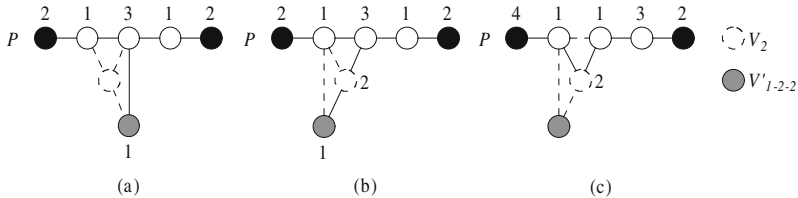


Fig. 1. (a) The first way. (b) The second way. (c) The last way.

We mark the vertices in $V'_{1-2-2} \cap P$ red to denote that these vertices have to be considered especially. Consider $P_i \in \mathbb{P}^{even}$ with at least one red vertex, we choose a vertex $v \in V'_{1-2-2}$ with $N(v) \cap P_i \neq \emptyset$ to do the third operation such that most red vertices in P_i can be located in even positions. We can choose v by scanning from left to right and right to left of P_i and record the position of each red vertex at the same time and then choose the best one at last. This cost $O(|P_i|)$ time. Since we have computed FVN to subtract $|\mathbb{P}^{even}|$ that must contain a division, this operation would not decrease FVN because it makes P increase 1 but \mathbb{P}^{even} decrease 1. Let P_1, P_2, \dots, P_s be the subpaths of P partitioned by black vertices. The following two lemmas show that the connecting order of V'_{1-2-2} will not affect the rank of the MVRST of G . Due to the limitation of pages, we omit the proof.

Lemma 8. *If each $P_i \notin \mathbb{P}^{even}$, then FVN is even.*

Lemma 9. *If each P_i contains odd number of vertices and $v, v' \in V'_{1-2-2}$, then connecting v or v' first will not affect the rank of MVRST.*

For our algorithm, we start to scan the vertices in P from left to right to connect V'_{1-2-2} . If a red vertex $u \in P$ which is adjacent to $v \in V'_{1-2-2}$ is in odd position of some P_i , then v can only apply the second or third connecting way. Otherwise, v can apply the first operation. We keep the position j to compute if u is in the even position of P_i . Note that the divisions are also computed in j . If u is in the odd position, we choose $w \in V_2$ which can connect to the most $v' \in V'_{1-2-2}$ and the vertex $u' \in P$ where u' is not in the $j(=4x)$ -th position in P , $x \in \mathbb{N}$, to apply the third operation with w . Note that one of the left and right vertices of u in P must be ranked more than 2. Clearly, if w does not exist, we can mark u white since each $v \in V'_{1-2-2}$ which is adjacent to u can connect to the vertex in the $4x$ -th position in P via some vertex in V_2 . Assume that w does exist. If $j - 1 \bmod 4 = 0$, w must be added to the right-hand side of u in P and each $v \in V'_{1-2-2}$ which can only connect to the right-hand side of u must be in $N(w)$. These vertices will be added to V_2 . Hence we can mark u white and scan the next vertex of u in P in position $j = j + 2$ since we had added w into P . If $j - 1 \bmod 4 \neq 0$, w must be added between u and the left-hand side vertex of u . Hence u is shifted to the even position and we do not need to do any operation on it. Then we scan the vertex of u in P in position $j = j + 2$. Note that we keep a variable to save that if there is a division before the next black vertex or red vertex which is in the even position that we do not want to change their position. Then we have to subtract this division from FVN after scanning a subpath. If $FVN = 0$ and there is at least one vertex $v \in V'_{1-2-2}$ which cannot do the first or second operation, we know that the rank limitation is broken. Hence we need to increase $rank(P)$. Thus we do *levelup*. The detail of our algorithm MVRST-Interval is as follows.

It is not hard to check that Algorithm MVRST-Interval can be implemented in linear time. Thus we have the following theorem.

Theorem 1. *MVRST problem on interval graphs can be solved in linear time.*

Procedure MVRST-Interval**Data:** An interval graph G .**Result:** A MVRST T of G with $rank(T)$.

1. Find the diameter P which contains v_{min} and v_{max} by BFS;
 2. **if** $|P| \leq 2^{\lfloor \log_2 |P| \rfloor} + 1$ **then**
 - └ done (i.e., $rank(T) = \lfloor \log_2 |P| \rfloor + 1$);
 3. Partition $V \setminus P$ into $V_1, V'_{1-1}, V'_{1-2-1}, V'_{1-2-2}$, and V_2 by BFS;
 4. Connect V_1 and V'_{1-1} to P and mark the connected vertices in P black;
 5. Partition P into P_1, P_2, \dots, P_s by the black vertices;
 - 5-1. Compute \mathbb{P}^{even} and $|D_s|$;
 - 5-2. $FVN = CMVRB - |P| - |\mathbb{P}^{even}| - |D_s|$;
 - 5-3. **if** $FVN < 0$ **then**
 - └ done (i.e., $rank = \lfloor \log_2 |P| \rfloor + 2$ and do *levelup*);
 - 5-4. **if** $V'_{1-2-2} = \emptyset$ **then**
 - └ done (i.e., $rank = \lfloor \log_2 |P| \rfloor + 1$);
 6. Mark the vertices in P which can be connected to V'_{1-2-2} red;
 - 6-1. **forall** $P_i \in \mathbb{P}^{even}$ **do**
 - └ Choose a red vertex in P_i to add the corresponding longest vertex $z \in V_2$ to P such that the most red vertices are in even positions of P_i and change the red vertex into white;
 - └ Move $N(z) \cap (V'_{1-1} \cup V'_{1-2-1} \cup V'_{1-2-2})$ to V_2 ;
 7. Scan the vertex u of P from left to right:
 - 7-1. **if** the position $j \bmod 2 \neq 0$ and u is red **then**
 - └ choose $w \in V_2$ such that $|N(w) \cap (N(u) \cap V'_{1-2-2})|$ is the largest and $\exists u' \in N(w) \cap (N(u) \cap P)$ such that u' is not in the $4x$ -th position in P ;
 - └ **if** w does not exist **then**
 - └ keep scanning the original righthand vertex in P ;
 - do the third operation by adding w to P ;
 - $FVN = FVN - 1$;
 - $j = j + 2$;
 - └ **if** $hidden_{division} = 1$ **then**
 - └ $hidden_{division} = 0$;
 - └ **else**
 - └ $hidden_{division} = 1$;
 - └ keep scanning the original righthand vertex in P ;
 - 7-2. **if** u is marked black or red and $j = 2x$ for some $x \geq 1$ and $hidden_{division} = 1$ **then**
 - └ $hidden_{division} = 0$;
 - └ $FVN = FVN - 1$;
 - └ $j = j + 2$;
 - 7-3. **if** $FVN < 0$ **then**
 - └ done (i.e., $rank = \lfloor \log_2 |P| \rfloor + 2$ and do *levelup*);
 - 8. Connect $a \in V_2$ to the even position in P and $b \in V'_{1-2-1}$ to the $4x$ -th position in P where $x \in \mathbb{N}$;
- Finally, we obtain T , (i.e., $rank(T) = \lfloor \log_2 |P| \rfloor + 1$);
- return** The spanning tree T and $rank(T)$;

3 Split Graphs

A Graph $G = (V, E)$ is called a *split graph* if its vertex set V can be partitioned into a clique K and an independent set I [4]. In this section, we will show that MVRST of a split graph can also be determined by its diameter.

Lemma 10. *The length of a diameter of a split graph G is at most 3.*

In the following, we let P_k denote a path with k vertices. It is easy to see that $\text{rank}(P_4) = 3$, $\text{rank}(P_3) = \text{rank}(P_2) = 2$, and $\text{rank}(P_1) = 1$. By Lemma 10, we know that the rank of MVRST of G is at least the rank of the diameter of G . In the case that the diameter of G is a P_1 , G contains only one vertex. If the diameter of G is a P_2 , then G is a clique. In this case, if $|V| > 2$, then the MVRST of G is a star; otherwise, G contains only two vertex. Thus the MVRST of G is easy to compute for these two cases. In the following, we only consider that $|V| \geq 3$. By Lemma 10, we only need to take care of the cases that the diameter of G is either a P_3 or a P_4 . We have the following lemmas.

Lemma 11. *Let $[u, v, w]$ be a diameter of a split graph G . Then v is a universal vertex of G and the MVRST of G is a star.*

Lemma 12. *Let $[u, v, x, y]$ be a diameter of a split graph G . Then a BFS tree T starting from v is a MVRST of G .*

According to Lemmas 11 and 12, we have the following theorem.

Theorem 2. *MVRST problem on split graphs can be solved in linear time.*

Corollary 1. *The rank of the MVRST of a split graph is equal to the rank of its diameter.*

4 Cographs

A graph G is called a *cograph* if it satisfies the following rules.

- Any single vertex graph is a cograph;
- If G is a cograph, so is its complement \overline{G} ;
- If G and H are cographs, so is their disjoint union $G \cup H$.

Let us consider a connected cograph. We first detect whether a universal vertex v exists or not. If v does exist, then we find the spanning tree T of G such that T is a star with v being the center. Otherwise, we do a BFS from an arbitrary vertex of G to obtain a BFS tree T .

Lemma 13. *Assume that there is no universal vertex in G . Then a BFS tree T is the MVRST of G and $\text{rank}(T) = 3$.*

Finally, according to Lemmas 13, we have the following theorem.

Theorem 3. *MVRST problem on cographs can be solved in linear time.*

5 Conclusion

In this paper, we show that the MVRST problem on interval graphs, split graphs, and cographs can be solved in linear time. Currently, only few results about this problem are known. The time complexity of the problem is still unknown for many classes of graphs. It seems that the diameter is important for this problem. In the future, we will study the problem on other classes of graphs, *e.g.*, chordal graphs and chordal bipartite graphs.

References

1. Bhattacharjee, A., Hasan, C.S., Kashem, M.A.: An Algorithm for Solving the Minimum Vertex Ranking Spanning Tree Problem on Series-parallel Graphs. In: 4th International Conference on Electrical and Computer Engineering, pp. 328–332 (2006)
2. Deng, H., Guha, S., Sen, A.: On a Graph Partition Problem with Application to VLSI Layout. *Information Processing Letters* 43, 87–94 (1992)
3. Greenlaw, R., Schäffer, A.A., de la Torre, P.: Optimal Edge Ranking of Trees in Polynomial Time. *Algorithmica* 13, 592–618 (1995)
4. Heggernes, P., Kratsch, D.: Linear-time Certifying Algorithms for Recognizing Split Graphs and Related Graph Classes. Technical Report (328), Department of Informatics, University of Bergen, Norway (2006)
5. Iyer, A.V., Ratliff, H.D., Vijayan, G.: Parallel Assembly of Modular Products—an Analysis. Technical Report, Georgia Institute of Technology (1988)
6. Iyer, A.V., Ratliff, H.D., Vijayan, G.: On Edge Ranking Problems of Trees and Graphs. *Discrete Applied Mathematics* 30, 43–52 (1991)
7. Leiserson, C.E.: Area Efficient Graph Layouts for VLSI. In: 21st Annual IEEE Symposium of Foundations of Computer Science, pp. 270–281 (1980)
8. Masuyama, S., Miyata, K., Nakayama, S., Zhao, L.: NP-hardness Proof and an Approximation Algorithm for the Minimum Vertex Ranking Spanning Tree Problem. *Discrete Applied Mathematics* 154, 2402–2410 (2006)
9. Masuyama, S., Nakayama, S.: An Algorithm for Solving the Minimum Vertex Ranking Spanning Tree Problem on Interval Graphs. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences* 86-A(5), 1019–1026 (2003)
10. Masuyama, S., Nakayama, S.: A Polynomial Time Algorithm for Obtaining a Minimum Vertex Ranking Spanning Tree in Outerplanar Graphs. *IEICE Transactions on Information and Systems* 89-D(8), 2357–2363 (2006)
11. Nevins, J., Whitney, D.: *Concurrent Design of Products and Processes*. McGraw-Hill, New York (1989)

On Constructions of Optimal Double-Loop Networks

Jianqin Zhou^{1,2,*} and Wenjuan Wang²

¹ Telecommunication School, Hangzhou Dianzi Univ., Hangzhou, 310018 China

² Computer Science School, Anhui Univ. of Technology, Ma'anshan, 243002 China
zhou9@yahoo.com

Abstract. A double-loop digraph $G(N; s_1, s_2)$ has N vertices $0, 1, \dots, N - 1$ and $2N$ edges of two types: s_1 -edge: $v \rightarrow v + s_1 \pmod{N}$, and s_2 -edge: $v \rightarrow v + s_2 \pmod{N}$, $v = 0, 1, \dots, N - 1$, for some fixed steps $1 \leq s_1 < s_2 < N$ with $\gcd(N; s_1, s_2) = 1$. Let $D(N; s_1, s_2)$ be the diameter of G and let us define: $D(N) = \min\{D(N; s_1, s_2) | 1 \leq s_1 < s_2 < N \text{ and } \gcd(N; s_1, s_2) = 1\}$. Given a fixed number of vertices N , the general problem is to find steps s_1 and s_2 , such that the digraph $G(N; s_1, s_2)$ has minimum diameter $D(N)$. A lower bound of this diameter is known to be $lb(N) = \lceil \sqrt{3N} \rceil - 2$. In this work, we give a simple and efficient algorithmic solution of the problem by using a geometrical approach. Given N , the algorithm find the minimum integer $k = k(N)$, such that $D(N) = lb(N) + k$. The running time complexity of the algorithm is $O(k^2)O(N^{1/4} \log N)$. With a new approach, we prove that infinite families of k -tight optimal double-loop networks can be constructed for any $k \geq 0$.

Keywords: Diameter; double-loop network; tight optimal; L -shaped tile; algorithm.

1 Introduction

Double-loop digraphs $G = G(N; s_1, s_2)$, with $1 \leq s_1 < s_2 < N$ and $\gcd(N; s_1, s_2) = 1$, have the vertex set $V = \{0, 1, \dots, N - 1\}$ and the adjacencies are defined by $v \rightarrow v + s_1 \pmod{N}$ and $v \rightarrow v + s_2 \pmod{N}$ for $v \in V$. The hops s_1 and s_2 between vertices are called steps. These kinds of digraphs have been widely studied as architecture for local area networks, known as double-loop networks (*DLN*).

The diameter of G is denoted by $D(N; s_1, s_2)$. As G is vertex symmetric, its diameter can be computed from the expression $\max\{d(0; i) | i \in V\}$, where $d(u; v)$ is the distance from u to v in G . For a fixed integer $N > 0$, the optimal value of the diameter is denoted by

$$D(N) = \min\{D(N; s_1, s_2) | 1 \leq s_1 < s_2 < N \text{ and } \gcd(N; s_1, s_2) = 1\}.$$

* The research is supported by Chinese Natural Science Foundation (No. 60473142) and Natural Science Foundation of Anhui Education Bureau of China(No. ZD2008005-1).

Several works studied the minimization of the diameter (for a fixed N) with $s_1 = 1$. Let us denote $D_1(N) = \min\{D(N; 1, s) | 1 < s < N\}$. A sharp lower bound for both $D(N)$ and $D_1(N)$ is well known to be $lb(N) = \lceil \sqrt{3N} \rceil - 2$.

A given $DLN G(N; s_1, s_2)$ is called k -tight if $D(N; s_1, s_2) = lb(N) + k (k \geq 0)$. A k -tight DLN is called optimal if $D(N) = lb(N) + k (k \geq 0)$, where integer N is called k -tight optimal. The 0-tight DLN are known as tight ones and they are also optimal. A given double-loop $G(N; 1, s)$ is called k -tight if $D(N; 1, s) = lb(N) + k (k \geq 0)$. A k -tight $DLN G(N; 1, s)$ is called optimal if $D_1(N) = lb(N) + k (k \geq 0)$.

The metrical properties of $G(N; s_1, s_2)$ are fully contained in its related L -shaped tile $L(N; l, h, x, y)$ where $N = lh - xy, l > y$ and $h \geq x$. In Figure 1, we illustrate generic dimensions of an L -shaped tile.

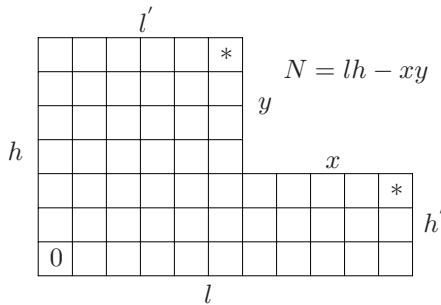


Fig. 1. Generic dimensions of an L -shaped tile

Let $D(L) = D(L(N; l, h, x, y)) = \max\{l + h - x - 2, l + h - y - 2\}$. For obvious reasons, the value $D(L)$ is called the diameter of the tile L . When an L -shaped tile $L(N; l, h, x, y)$ has diameter $lb(N) + k$, we say it is k -tight.

Given an N , it is desirable to find a double-loop network $G(N; s_1, s_2)$ with its diameter being equal to $d(N)$. Aguiló and Fiol in [1] gave an algorithm to search an L -shaped tile $L(N; l, h, x, y)$ with diameter $\lceil \sqrt{3N} \rceil - 2 + k$ in the order $k = 0, 1, 2, \dots$. The first-found L -shaped tile must have minimum diameter, but no explicit algorithm was given in their paper. They estimated the time complexity of this algorithm to be $O(k^3)O(\log N)$. However, later we will show that the time complexity of their algorithm is $O(k^2)O(N^{1/4} \log N)$. An algorithm was given in [4] to find optimal double-loop networks with non-unit steps.

In this paper, we propose a simple and efficient algorithm to search an L -shaped tile $L(N; l, h, x, y)$ with diameter $\lceil \sqrt{3N} \rceil - 2 + k$ in the order $k = 0, 1, 2, \dots$. Our algorithm is based on some theorems of Li and Xu [7,8]. Unlike the algorithm of Aguiló and Fiol, our algorithm does not require complex formulas. The running time complexity of our algorithm is $O(k^2)O(N^{1/4} \log N)$.

It is proved in [3] that for any $k \geq 0$, there exist infinite families of k -tight optimal double-loop networks. With a new approach, we prove that infinite families of k -tight optimal double-loop networks can be constructed for any $k \geq 5$.

2 Preliminary

The following Lemma 1, 2, 3, and 4 can be found in [6 or 7].

Lemma 1^[6]. Let t be a nonnegative integer. We define $I_1(t) = [3t^2 + 1, 3t^2 + 2t]$, $I_2(t) = [3t^2 + 2t + 1, 3t^2 + 4t + 1]$ and $I_3(t) = [3t^2 + 4t + 2, 3(t + 1)^2]$. Then we have $[4, 3T^2 + 6T + 3] = \bigcup_{t=1}^T \bigcup_{i=1}^3 I_i(t)$, where $T > 1$, and $lb(N) = 3t + i - 2$ if $N \in I_i(t)$ for $i = 1, 2, 3$.

Lemma 2^[7]. Let $L(N; l, h, x, y)$ be an L -shaped tile, where $N = lh - xy$, $l > y$ and $h \geq x$. Then

- (a) There exists $G(N; 1, s)$ realizing the L -shaped tile iff $\gcd(h, y) = 1$, where $s \equiv \alpha l - \beta(l - x) \pmod{N}$ for some integral values α and β satisfying $\alpha y + \beta(h - y) = 1$.
- (b) There exists $G(N; s_1, s_2)$ realizing the L -shaped tile iff $\gcd(l, h, x, y) = 1$, where $s_1 \equiv \alpha h + \beta y \pmod{N}$, $s_2 \equiv \alpha x + \beta l \pmod{N}$ for some integral values α and β satisfying $\gcd(N, s_1, s_2) = 1$.

Lemma 3^[7]. Let $L(N; l, h, x, y)$ be an L -shaped tile, $N = lh - xy$. Then

- (a) If $L(N; l, h, x, y)$ is realizable, then $|y - x| < \sqrt{N}$;
- (b) If $x > 0$ and $|y - x| < \sqrt{N}$, then
$$D(L(N; l, h, x, y)) \geq \sqrt{3N - \frac{3}{4}(y - x)^2} + \frac{1}{2}|y - x| - 2;$$
- (c) Let $f(z) = \sqrt{3N - \frac{3}{4}z^2} + \frac{1}{2}z$. Then $f(z)$ is strictly increasing when $0 \leq z \leq \sqrt{N}$.

Lemma 4^[7]. Let $N(t) = 3t^2 + At + B \in I_i(t)$ and L be the L -shaped tile $L(N(t); l, h, x, y)$, where A and B are integral values; $l = 2t + a$, $h = 2t + b$, $z = |y - x|$, a, b, x, y are all integral polynomials of variable t , and $j = i + k (k \geq 0)$. Then L is k -tight iff the following identity holds,

$$(a + b - j)(a + b - j + z) - ab + (A + z - 2j)t + B = 0. \tag{1}$$

The following Lemma 5 is the generalization of Theorem 2 in [8], and can be found in [9].

Lemma 5^[9]. Let $H(z, j) = (2j - z)^2 - 3[j(j - z) + (A + z - 2j)t + B]$, and the identity (1) be an equation of a and b . A necessary condition for the equation (1) to have integral solution is that $4H(z, j) = s^2 + 3m^2$, where s and m are integers.

It is easy to show that the following Lemma 6 is equivalent to Theorem 1 in [8]. Lemma 6 can be found in [9].

Lemma 6^[9]. Let n, s and m be integers, $n = s^2 + 3m^2$. If n has a prime factor p , where $p \equiv 2 \pmod{3}$, then there exists an even integer q , such that n is divisible by p^q , but not divisible by p^{q+1} .

The following lemma is the generalization of Lemma 14 in [3].

Lemma 7. For given integer $k \geq 5$, let $N = 3t^2 + At + k^2 + k - i(i - 1) \in I_i(t)$, where $A = 2i - 1, 1 \leq i \leq 3$.

Case 1. Let L be k_0 -tight L -shaped tile $L(N; l, h, x, y), z = |y - x|$, where $0 \leq k_0 < k$. Then $z \leq 2k_0$.

Case 2. Let L be k -tight L -shaped tile $L(N; l, h, x, y), z = |y - x|$. Then $z \leq 2k + 1$.

Lemma 8^[9]. There exists an infinite number of prime p , where $p \neq 2$ and $p \equiv 2 \pmod{3}$.

3 The Algorithm of Aguiló and Fiol

In general, the algorithm for searching a k -tight double loop digraph works as follows:

Algorithm 1^[1]: to search k -tight L -shaped tiles with a given N .

Step 1. Given N , find $t, lb(N)$ and $i \in \{1, 2, 3\}$ such that $N \in I_i(t)$; Assign $k = 0$.

Step 2. For a fixed k and from $z = 0$ to $z = \lfloor z_{\max} \rfloor$, where z_{\max} is obtained by a complex function about N and k , look for a k -tight tile $L(l, h, x, y)$ with $y = x + z$ and $\gcd(l, h, x, y) = 1$. The following is the order of searching:

- A. Procreating case: $z \in \{2k, 2k + 1, 2k + 2\}$.
- B. $E_2(k)$ case: the variation of z is recommended to be from $2k - 1$ to 0 .
- C. $E_1(k)$ case: the variation of z is recommended to be from $2k + 3$ to $\lfloor z_{\max} \rfloor$.

Step 3. If an L -shaped tile has been found in Step 2, then there exists a k -tight (optimal) digraph with N vertices and steps s_1, s_2 such that $D(N; s_1, s_2) = D(N) = lb(N) + k$, and the algorithm ends. Otherwise there is no such digraph with such a k . Then, assign $k \leftarrow k + 1$ and go to Step 2.

Step 2 is the most complex one. For instance, in the $E_2(k)$ case we must search for integral points in the plane which are on the ellipsis

$$EB_i = ab - (a + b - k - i)(a + b + k - i - s), \text{ for } i = 1, 2, 3 \text{ and } 1 \leq s \leq 2k.$$

One integer pair (a, b) will be valid if $b \in [b_i^-, b_i^+]$ and $a = a_i^-$ or $a = a_i^+$. These values can be obtained as follows.

$$\begin{aligned} b_i^\pm &= \frac{s+2i \pm 2\sqrt{m_i}}{3}, \\ m_i &= (s + 2i)^2 + 3[(k + i)(k - s - i) - EB_i], \\ a_i^\pm &= \frac{s+2i - b \pm \sqrt{q_i}}{2}, b \in [b_i^-, b_i^+], \\ q_i &= -3b^2 + 2(s + 2i)b + (s + 2i)^2 + 4[(k + i)(k - s - i) - EB_i]. \end{aligned}$$

Consider $N = 3t^2 + (s + 2i)t + EB_i$, for $i = 1, 2, 3$ and $1 \leq s \leq 2k$, from Lemma 1, we know that $EB_1 \in [-2t + 1, 0], EB_2 \in [-2t + 1, 1]$ and $EB_3 \in [-2t + 2, 3]$.

Note that $t = O(N^{1/2})$, so the computing cost for searching integral points is $O(k^2)O(N^{1/4})$. However, Aguiló and Fiol [1] estimated the computing cost to be $O(k^3)$. The numerical computations and the results of Coppersmith reported in [5] suggest that the order of k might be as low as $O(\log^{1/4} N)$. Based on our experiments, later we will show that k might be less than $O(\log N)$.

To check that $\gcd(l, h, x, y) = 1$, it is well-known that the order of the Euclidean algorithm to compute such a gcd is $O(\log N)$.

Chan, Chen and Hong [2] proposed a simple algorithm to find s_1 and s_2 for a given L -shaped tile. The algorithm takes at most $O((\log N)^2)$ time. However, note that this step is only executed when a solution has been found. Therefore, its cost has no relevance to the total order of the algorithm. As a conclusion, the algorithm has order $O(k)O(N^{1/4} \log N)$ if k is given and $O(k^2)O(N^{1/4} \log N)$ otherwise.

4 The Algorithm to Search L -Shaped Tiles

We now prove that

Theorem 1. Let $N = 3t^2 + At + B \in I_i(t), D(N) = lb(N) + k(k \geq 0)$ and L be k -tight L -shaped tile $L(N; l, h, x, y); z = |y - x|$. Then the following holds

Case 1. If $A = 0$ or $A = 2(i = 2)$ or $A = 4(i = 3)$, and $3N - \frac{3}{4}(2k + 3)^2 > (3t + \frac{A-1}{2})^2$, then $0 \leq z \leq 2k + 2$.

Case 2. If $A = 1$ or $A = 3$ or $A = 5$, and $3N - \frac{3}{4}(2k + 2)^2 > (3t + \frac{A-1}{2})^2$, then $0 \leq z \leq 2k + 1$.

Case 3. If $A = 2(i = 1)$ or $A = 4(i = 2)$ or $A = 6$, and $3N - \frac{3}{4}(2k + 1)^2 > (3t + \frac{A-1}{2})^2$, then $0 \leq z \leq 2k$.

Proof. We only prove Case 1, the others being similar.

Let $L(N; l, h, x, y)$ be k -tight, then $D(N) = 3t + i - 2 + k$. Note that $i = A/2 + 1$. By lemma 3, if $z \geq 2k + 3$, we have

$$\begin{aligned} D(L(N; l, h, x, y)) &\geq \sqrt{3N(t) - \frac{3}{4}(2k + 3)^2} + \frac{2k+3}{2} - 2 \\ &> (3t + \frac{A-1}{2}) + \frac{2k+3}{2} - 2 \\ &= 3t + i - 2 + k \\ &= lb(N) + k. \end{aligned}$$

Therefore, all the k -tight L -shaped tile $L(N; l, h, x, y)$ must satisfy $0 \leq z \leq 2k + 2, z = |y - x|$. We have this theorem. □

Based on computer search, we know that

Theorem 2. For $N \leq 10^8$, if N with k -tight optimal double loop digraph, then $k < \log_{10} N$.

We continue to consider the case 1 of Theorem 1, the others being similar.

For $N = 3t^2 + At + B \in I_i(t), 3N - \frac{3}{4}(2k + 3)^2 > (3t + \frac{A-1}{2})^2$ is equal to $3t + 3B > \frac{3}{4}(2k + 3)^2 + \frac{1}{4}(A - 1)^2$.

For $33 \leq t < 372$, it is known that $k \leq 4$. For $A = 0, 2, 4$,

$$\begin{aligned} \frac{3}{4}(2k + 3)^2 + \frac{1}{4}(A - 1)^2 &< \frac{3}{4}(2 \times 4 + 3)^2 + \frac{1}{4}(3)^2 \\ &= 93 < 3 \times 33 + 3 \\ &\leq 3t + 3B. \end{aligned}$$

For $372 \leq t < 6000$, it is known that $k \leq 7$. For $A = 0, 2, 4$,

$$\begin{aligned} \frac{3}{4}(2k + 3)^2 + \frac{1}{4}(A - 1)^2 &< \frac{3}{4}(2 \times 7 + 3)^2 + \frac{1}{4}(3)^2 \\ &= 219 < 3 \times 372 + 3 \\ &\leq 3t + 3B. \end{aligned}$$

Therefore, $0 \leq z \leq 2k + 2$ holds for $0 \leq t \leq 6000$. For $t < 33$, we can directly verify that $0 \leq z \leq 2k + 2$ holds.

In fact, note that $t = O(N^{1/2})$, t increases much faster than k^2 does, hence $3t + 3B > \frac{3}{4}(2k + 3)^2 + \frac{1}{4}(A - 1)^2$, thus $0 \leq z \leq 2k + 2$ may holds for $t \geq 0$.

Based on Theorem 2, we have the following conclusion.

Theorem 3. For $N \leq 10^8$, let $N = 3t^2 + At + B \in I_i(t)$, where $t = \lceil \sqrt{N/3} \rceil - 1$, $A = \lfloor (N - 3t^2)/t \rfloor$, $B = N - 3t^2 - At \geq 0$. If $D(N) = lb(N) + k(k \geq 0)$ and L be k -tight L -shaped tile $L(N; l, h, x, y)$; $z = |y - x|$. Then the following holds

- Case 1. If $A = 0$ or $A = 2(i = 2)$ or $A = 4(i = 3)$, then $0 \leq z \leq 2k + 2$;
- Case 2. If $A = 1$ or $A = 3$ or $A = 5$, then $0 \leq z \leq 2k + 1$;
- Case 3. If $A = 2(i = 1)$ or $A = 4(i = 2)$ or $A = 6$, then $0 \leq z \leq 2k$.

The numerical computations and the results of Coppersmith reported in [5] suggest that the order of k might be as low as $O(\log^{1/4} N)$. Therefore, Theorem 2 and Theorem 3 might be true for $N > 10^8$.

From Theorem 3, it does not need to compute z_{\max} for every N and k , and it does not need to consider the $E_1(k)$ case, where the variation of z is from $2k + 3$ to $\lfloor z_{\max} \rfloor$.

We can now describe our algorithm to search k -tight double loop networks based on Theorem 3.

Algorithm 2: to search k -tight L -shaped tiles with a given N .

Step 1. Given N , calculate $t = \lceil \sqrt{N/3} \rceil - 1$, $A = \lfloor (N - 3t^2)/t \rfloor$, $B = N - 3t^2 - At \geq 0$, $lb(N) = \lceil \sqrt{3N} \rceil - 2$, $i = lb(N) - 3t + 2$, $k = 0$.

- Case 1. If $A = 0$ or $A = 2(i = 2)$ or $A = 4(i = 3)$, then $z_0 = 2$;
- Case 2. If $A = 1$ or $A = 3$ or $A = 5$, then $z_0 = 1$;
- Case 3. If $A = 2(i = 1)$ or $A = 4(i = 2)$ or $A = 6$, then $z_0 = 0$.

Step 2. For a fixed k and from $z = 0$ to $z = 2k + z_0$, look for k -tight tile $L(l, h, x, y)$. The following is the order of searching:

```

Loop: While(TRUE)
    j = i + k;
    Loop: for(z = 0; z < 2k + z0 + 1; z = z + 1)
        If the following equation
    
```

$(a + b - j)(a + b - j + z) - ab + (A + z - 2j)t + B = 0$
 has integral solution (a, b) ,
 and $\gcd(l, h, x, y) = 1, l > y$ and $h \geq x$,
 where $l = 2t + a, h = 2t + b, x = t + a + b - j, y = t + a + b - j + z$
 then break these two loops.

End for
 $k = k + 1;$

End While

Step 3. Then there exists a k -tight(optimal) digraph with N vertices and steps s_1, s_2 such that $D(N; s_1, s_2) = D(L(l, h, x, y)) = D(N) = lb(N) + k$, and the algorithm ends.

For the equation: $(a + b - j)(a + b - j + z) - ab + (A + z - 2j)t + B = 0$, it can be changed to the following

$$\frac{a^2 + b^2}{2} + \left(\frac{a + b}{\sqrt{2}} + \frac{-2j + z}{\sqrt{2}}\right)^2 = \frac{(-2j + z)^2}{2} + j(z - j) + (2j - z - A)t - B.$$

Note that $j = i + k$, and $0 \leq z \leq 2k + 2$, hence $\frac{(-2j+z)^2}{2} \leq 2j^2, j(z - j) \leq j(k + 2 - i)$, so $b^2 \leq 2[2j^2 + j(k + 2 - i) + (2j - A)t - B]$.

To solve the equation, we only need to check whether the quadratic equation $(a + b - j)(a + b - j + z) - ab + (A + z - 2j)t + B = 0$ has integral solution a , where $b \in [-b_0, b_0], b_0 = 2^{1/2}[2j^2 + j(k + 2 - i) + (2j - A)t - B]^{1/2}$.

Note that $t = O(N^{1/2}), B = O(N^{1/2})$, hence $b = O(N^{1/4})$, so the computing cost for searching integral points (a, b) is $O(k^2)O(N^{1/4})$.

To check that $\gcd(l, h, x, y) = 1$, it is well-known that the order of the Euclidean algorithm to compute such a gcd is $O(\log N)$. Therefore, the computing cost for searching k -tight L -shaped tiles, which can realized by $G(N; s_1, s_2)$, is $O(k^2)O(N^{1/4} \log N)$.

If $D(N) = lb(N) + k(k \geq 0)$, then there exist k -tight L -shaped tiles, from Lemma 4 and Lemma 2, equation (1) has integral solution (a, b) , such that $L(N; l, h, x, y)$ or $L(N; h, l, y, x)$ can be realized by $G(N; s_1, s_2)$, where $l = 2t + a, h = 2t + b, x = t + a + b - j, y = t + a + b - j + z$. From Theorem 3, it is only need to search k -tight L -shaped tiles for $0 \leq z \leq 2k + z_0$, where z_0 obtained from A and i . Therefore, Algorithm 2 is correct.

5 Infinite Families of k -Tight Optimal Double-Loop Networks

For $0 \leq k \leq 4$, from [8,9], there exist infinite families of k -tight optimal double-loop networks. Thus we consider that $k \geq 5$.

Theorem 4. For given integer $k \geq 5$, let $N(t) = 3t^2 + At + k^2 + k - i_0(i_0 - 1) \in I_{i_0}(t)$, where $A = 2i_0 - 1, 1 \leq i_0 \leq 3$ and $\gcd(2, i_0 + k) = 1$. Then infinite families of k -tight optimal double-loop network can be constructed.

Proof. We prove that $4H(z, j)$ has a prime factor p with an odd power, where $p \equiv 2 \pmod 3$, $0 \leq z \leq 2k_0, j = i_0 + k_0, 0 \leq k_0 \leq k$. Since $H(z, j) = (2j - z)^2 - 3[j(j - z) + (A + z - 2j)t + B], A + z - 2j \leq 2i_0 - 1 + 2k_0 - 2(i_0 + k_0) = -1$, hence $4H(z, j)$ are all polynomials of order 1. Let us denote these polynomials by: $a_i t + b_i, 1 \leq i \leq d$, where $d = (k + 1)^2$.

From Lemma 8, let us denote all primes by: $p_1, p_2, \dots, p_i, \dots$, where $p_i \equiv 2 \pmod 3, \gcd(p_i, 2) = 1$ and $\gcd(p_i, i_0 + k) = 1$ for $i \geq 1$.

Without loss of generality, we assume that $\gcd(a_i, p_i) = 1$ for $1 \leq i \leq d$. Suppose $\alpha_i a_i + \beta_i p_i^2 = 1$, then we have $\alpha'_i a_i + \beta'_i p_i^2 = p_i - b_i$, that is, $\alpha'_i a_i + b_i = -\beta'_i p_i^2 + p_i$. Thus there exists c_i , such that $a_i t + b_i \equiv p_i \pmod{p_i^2}$ for any $t = p_i^2 e + c_i, e \geq 0$.

Since p_i^2 mutually prime to each other, by Chinese Remainder Theorem, we know that there exists a solution to the following congruences.

$$\begin{cases} t - i_0 - k \equiv 1 \pmod 2 \\ t \equiv 1 \pmod{i_0 + k} \\ t \equiv c_1 \pmod{p_1^2} \\ t \equiv c_2 \pmod{p_2^2} \\ \dots\dots \\ t \equiv c_d \pmod{p_d^2} \end{cases}$$

Suppose the solution is $t = 2(i_0 + k)p_1^2 p_2^2 \dots p_d^2 e + c, e \geq 0$. If $0 \leq z \leq 2k_0, j = i_0 + k_0, 0 \leq k_0 \leq k, t = 2(i_0 + k)p_1^2 p_2^2 \dots p_d^2 e + c$, then $4H(z, j)$ has a prime factor p with an odd power, where $p \equiv 2 \pmod 3$. By Lemma 7, 6, 5, and 4, we know that there does not exist any k_0 -tight L -shaped tile $L(l, h, x, y)$.

We now prove that there exists k -tight L -shaped tile $L(l, h, x, y)$, where $z = 2k + 1$. Consider $j = i + k$ and $z = 2k + 1$, equation (1) is equivalent to the following

$$(a + b)^2 - (2i - 1)(a + b) - ab = 0.$$

$(0, 0)$ is a solution of the equation. We have $l = 2t + a = 2t, h = 2t + b = 2t, y = t + a + b - j = t - i_0 - k, x = y + z = t - i_0 + k + 1$.

Thus $\gcd(h, y) = \gcd(2t, t - i_0 - k) = \gcd(2(i_0 + k), t - i_0 - k) = 1$. From Lemma 2, there exist infinite families of k -tight optimal double-loop networks $G(N(t); 1, s)$, where $t = 2(i_0 + k)p_1^2 p_2^2 \dots p_d^2 e + c$. We have this theorem. \square

References

1. Aguiló, F., Fiol, M.A.: An Efficient Algorithm to Find Optimal Double Loop Networks. *Discrete Mathematics* 138, 15–29 (1995)
2. Chan, C.F., Chen, C., Hong, Z.X.: A Simple Algorithm to Find the Steps of Double-loop Networks. *Discrete Applied Mathematics* 121, 61–72 (2002)
3. Chen, B., Chen, X., Meng, J., Xiao, W.: The Construction of Infinite Families of Any K -tight Optimal and Singular K -tight Optimal Directed Double Loop Networks. *Science in China, Series A* 50(7), 1055–1064 (2007)
4. Dai, X., Zhou, J., Wang, K.: An Algorithm to Find Optimal Double-loop Networks with Non-unit Steps. In: Xu, M., Zhan, Y., Cao, J., Liu, Y. (eds.) APPT 2007. LNCS, vol. 4847, pp. 349–358. Springer, Heidelberg (2007)

5. Erdős, P., Hsu, D.F.: Distributed Loop Networks with Minimum Transmission Delay. *Theoret. Comput. Sci.* 100, 223–241 (1992)
6. Esqué, P., Aguiló, F., Fiol, M.A.: Double Commutative-step Digraphs with Minimum Diameters. *Discrete Mathematics* 114, 147–157 (1993)
7. Li, Q., Xu, J., Zhang, Z.: The Infinite Families of Optimal Double Loop Networks. *Discrete Applied Mathematics* 46, 179–183 (1993)
8. Xu, J., Liu, Q.: An Infinite Family of 4-tight Optimal Double Loop Networks. *Science in China, Series A* 46(1), 139–143 (2003)
9. Zhou, J.: K-tight Double Loop Networks and Their Infinite Families. *Acta Mathematica Sinica, Chinese Series* 48(6), 1213–1220 (2005)

Research on Resource Selection with Precedence and Due Date Constraint*

Shujuan Li¹, Yong Liu¹, Yan Li¹, Yuefei Xu², and Zhibin Zeng¹

¹ School of mechanical & instrumental engineering,
Xi'an University of Technology, 710048 Xi'an, PRC

² IEEE member, 6833 Station Hill Dr. Vancouver,
British Columbia, Canada, V3N5E1
shujuanli@xaut.edu.cn

Abstract. In this paper, we consider the resource selection problem with due date constraint in production planning and control where the subtasks form a precedence network. We prove that this problem is NP-complete. Thus this problem cannot have any polynomial time solution algorithm at present. We establish a nonlinear integer-programming model for this problem, and prove the monotonicity properties of the objective function and constraint function in the model. Basing on our observations, we construct a Branch & Bound method to solve the problem. The test results show that this algorithm is effective.

1 Introduction

In production plan and controlling process, it is usually happened that a task can be separated into several subtasks, and then the bid invitation and bidding or other means are adopted to select the best resources to undertake each subtask separately, so that the entire task will be completed simultaneously. Thus, the resource selection problem is one of the research focuses on production scheduling^[1, 2].

In this study, such factors affecting resource selection as the entire costs, the completion time and the precedence relationship among subtasks etc are taken into consideration. Talluri, S proposed a two-phase mathematical programming approach for resource selection in which the factors of cost and completion time were considered^[3]. Largely owing to the precedence relationship of subtasks, the entire task can be represented by an activity network diagram with precedence (PERT diagram)^[4], whereby the precedence relationship of subtasks is considered first in studying resource selection problem. The integer programming model and genetic algorithm are used to solve the partner selection problem in considering the entire expenses of task, completion time and the precedence relationship of subtasks in co-operation enterprise alliance in literature^[5]. But the complexity of problem is not proven in literature^[3, 5]. The bid invitation and bidding methods in contract net and mediator are adopted in resource selection with due date constraint in literature^[6]. However, when this method

* This work was supported by the Natural Science Fund of ShaanXi Province (Grant No. 2004E202) and Chunhui Planning of the Education Ministry of China (Grant No. Z2005-1-61004).

is adopted to carry out resource selection, and at same time when the magnitude of the problem increases to a certain extent, the search speed of this algorithm will be greatly reduced and even no optimal solution can be found.

Generally speaking, the due date of task delivery can not be delayed. With the production scheduling process as the object, and the production cost as the goal as well as the precedence relation and delivery date of the subtask as the constraints, this paper deals with the integer programming method and the Branch & Bound optimal in order to solve the problem of resource selection. Accordingly, this problem can be expressed as a non-linear integer programming, also, a 0-1 Knapsack problem is used to prove that problem which a NP-complete problem. Therefore, this paper makes a further analysis of characteristics of this problem on the basis of which a Branch & Bound algorithm is devised. In contrast with the bid invitation and bidding algorithm, the branch and bound method is superior to the settlement of the increasing magnitude of the problem in selecting resource as quickly as possible.

In this paper, we consider a simplified resource selection problem with due date constraint which takes into account only the bid cost and the bid completion time of subtasks, and the due date of the task. We model the problem as a nonlinear integer programming problem and prove that the decision problem of the resource selection problem with due date constraint is NP-complete in section 2. Then, we analyze some properties of the resource selection problem with due date constraint and construct a Branch & Bound algorithm in section 3. Numerical experiments are included in section 4. Finally, the concluding remarks are presented in section 5.

2 Model and Complexity of the Problem

Suppose that a main contractor wins a big task which consists of several subtasks, which form a precedence activity network. An example of a task consisting of 13 subtasks is shown in Fig. 1.

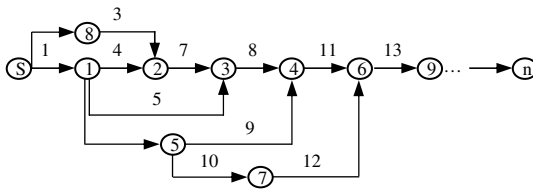


Fig. 1. Example of a task represented by an activity network

If subtask k can only begin after the completion of subtask i , H is an activity network that constructs the precedence of these subtasks. i.e., subtask i precedes subtask k , then we define the connected subtask pair by $\langle i, k \rangle \in H$. We label these subtasks such that $i < k$. The final subtask is labeled as subtask n . We create a virtual final subtask to follow subtasks 11 and 12, and label it as subtask 13. Thus, we can define that the completion time of final subtask C_n is the completion time of the task.

For subtask $i, i = 1, 2, \dots, n$, suppose that there are m_i resource agents responding to the tender invitation. To simplify the problem, we assume that the main contractor will select only one candidate resource agent to undertake one subtask. The objective of the main contractor is to select the optimal resource agents for all subtasks to minimize cost of the task within due date.

Let $x = (x_1, x_2, \dots, x_n)^t$ be a solution of the problem, where x_i is the resource agent which win a tender for subtask i . So the total cost of the task is $\sum_{i=1}^n b_{ix_i}$ and the solution space of the problem is $S = D_1 \times D_2 \times \dots \times D_n$.

If x is a solution of the problem, let $S_i(x)$ and $C_i(x)$ express beginning time and completion time of subtask i separate, then $C_n(x)$ can be worked out through the critical path method. So the problem can be described as follows:

$$(P) \begin{cases} \min & f(x) = \sum_{i=1}^n b_{ix_i} \\ \text{s.t.} & C_{nx_n} \leq D, x = (x_1, x_2, \dots, x_n)^T \in S = D_1 \times \dots \times D_n \end{cases} \quad (1)$$

Obviously, problem (P) is a non-linear integer programming problem, where $C_{nx_n} \leq D$ is not expressed explicitly, for a solution x , the satisfaction of the constraint can be tested. This problem can be formulated as follow:

Instance 1: given an activity network H , the resource agent x_i of subtask i is $x_i \in d_i = \{1, 2, \dots, m_i\}$. The bid cost b_{ix_i} and the bid completion time q_{ix_i} of x_i for subtask i are a rational number and an integer, respectively, the due date d of the task is an integer. B is a rational number.

Question: does there exist $x_i \in d_i, i = 1, 2, \dots, n$, which satisfy that $C_{nx_n} \leq D$ and $f(x) \leq B$?

Theorem 1. The decision problem of the resource selection problem is a NP-complete problem.

Proof. Given a solution x of instance 1, we can find a critical path of the activity network H , and the length of the critical path is C_{nx_n} . Thus we can judge in polynomial time whether C_{nx_n} is less than or equal to D , calculate $f(x)$ and judge whether it is less than or equal to B . So the decision problem of the resource selection problem is a NP-complete problem. Next we prove that the 0-1 Knapsack problem is a sub-problem of the resource selection problem by the restriction method.

Suppose that the activity network H is shown as in Fig.2.

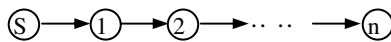


Fig. 2. An activity network H

Let $m_1 = m_2 = \dots = m_n = 2$, namely, there are only two resource Agents (candidates) for each subtask, and let $q_{i1} = 0, b_{i1} = c_i, q_{i2} = t_i, b_{i2} = 0$, where c_i, t_i are integers, $i = 1, 2, \dots, n$.

Since the network in Fig.2 is linear precedent, the critical path of the activity network H is the unique path from s to n . Therefore the resource selection problem can be formulated as:

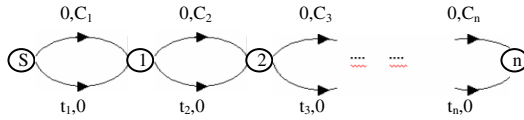


Fig. 3. A directed graph

Does there exist a path in Fig.3 from s to n which makes the summation of the first numbers of all edges in the path less than or equal to D and the summation of the second numbers of all edges in the path less than or equal to B ?

In Fig.3 at stage i , if we select the upper edge as an edge of a path from s to n , then let $x_i = 0$, and if we select the lower edge as an edge of a path from s to n , then let $x_i = 1, i = 1, 2, \dots, n$. Thus the summation of the first numbers of all edges in the path from s to n is $t_1x_1 + t_2x_2 + \dots + t_nx_n$, and the summation of the second numbers of all edges in the path is

$$c_1(1 - x_1) + c_2(1 - x_2) + \dots + c_n(1 - x_n) = \sum_{i=1}^n c_i - (c_1x_1 + c_2x_2 + \dots + c_nx_n)$$

So the decision problem of the particular type of the resource selection problem can be modeled as:

Instance 2: given nonnegative integers $t_i, c_i, i = 1, 2, \dots, n$, and nonnegative integers D, B .

Question: does there exist $x_i \in \{0, 1\}, i = 1, 2, \dots, n$, such that

$$t_1x_1 + t_2x_2 + \dots + t_nx_n \leq D \text{ and } \sum_{i=1}^n c_i - (c_1x_1 + c_2x_2 + \dots + c_nx_n) \leq B ?$$

So instance 2 can be rewritten as follows.

Instance 3: given nonnegative integers $t_i, c_i, i = 1, 2, \dots, n$, and nonnegative integers D, B .

Question: does there exist $x_i \in \{0, 1\}, i = 1, 2, \dots, n$, such that

$$t_1x_1 + t_2x_2 + \dots + t_nx_n \leq D, \text{ and } c_1x_1 + c_2x_2 + \dots + c_nx_n \geq \sum_{i=1}^n c_i - B ?$$

It is obvious that instance 3 is the decision problem of the 0-1 Knapsack problem. Hence, the decision problem of the resource selection problem is NP-complete problem.

3 Properties of the Problem and Branch and Bound Method

In section 2, we have proved that the NP-completeness of the resource selection problem, so the problem cannot be solved by any polynomial time algorithm up to now. Hence we can only find a solution of the problem by the Branch & Bound methods or heuristic methods like genetic algorithm^[6] or the filled function method^[7]. Generally speaking, the main contractor hopes to find an optimal solution. So we construct a Branch & Bound method for the problem in this section.

From the practical issues, if for two resource agents j and k of subtask i , if $b_{ik} \leq b_{ij}$ and $q_{ik} < q_{ij}$, or $b_{ik} < b_{ij}$ and $q_{ik} \leq q_{ij}$, then it is easy to prove theoretically that all optimal solutions of problem (P) does not include contractor j , so we can weed out resource agent j in the input data. Suppose that

$$b_{i1} > b_{i2} > \dots > b_{im}, q_{i1} < q_{i2} < \dots < q_{im}, i = 1, 2, \dots, n \tag{2}$$

Now we give two notations, for two solutions x and y of problem (P), notation $x \leq y$ means that $x_i \leq y_i, i = 1, 2, \dots, n$; notation $x < y$ means that $x_i \leq y_i, i = 1, 2, \dots, n$ and there exists at least one subscript j such that $x_j < y_j$.

Theorem 2. $F(x)$ in problem (P) is a strictly monotonically decreasing function, and C_{nx_n} is a monotonically increasing function of x .

Definition 1. Suppose that $L = (l_1, l_2, \dots, l_n)^T \leq U = (u_1, u_2, \dots, u_n)^T$, and $l_i, u_i, i = 1, 2, \dots, n$ are integers. Then $X = [L, U] = \{x \in R^n : l_i \leq x_i \leq u_i, i = 1, 2, \dots, n\}$ is one box in R^n . Thus problem (P) can be rewritten as:

$$\begin{cases} \min & f(x) = \sum_{i=1}^n b_{ix_i} \\ \text{s.t.} & C_{nx_n} \leq D, x = (x_1, x_2, \dots, x_n)^T \in [E, M] \cap I^n \end{cases} \tag{3}$$

Where $E = (1, 1, \dots, 1)^T, M = (m_1, m_2, \dots, m_n)^T, I^n$ is the set of integer points in R^n .

When we solve problem (3) with a Branch & Bound method, we need to divide one box into two-sub boxes. Adoptive method is shown as follows:

Suppose that the box needed to be divided is $X = [L, U]$. Initially $X = [E, M]$. Find a subscript i' such that $u_{i'} - l_{i'} = \max_{i=1, 2, \dots, n} (u_i - l_i)$, and divided X into two sub-boxes as follows:

$$X_1 = \left\{ x \in X : x_{i'} \geq l_{i'} + \left\lfloor \frac{u_{i'} - l_{i'}}{2} \right\rfloor + 1 \right\}, X_2 = \left\{ x \in X : x_{i'} \geq l_{i'} + \left\lfloor \frac{u_{i'} - l_{i'}}{2} \right\rfloor \right\},$$

Where $\left\lfloor \frac{u_{i'} - l_{i'}}{2} \right\rfloor$ is the maximum integer less than or equal to $\frac{u_{i'} - l_{i'}}{2}$.

Theorem 3. Suppose that $X = [L, U]$ is one of the sub-boxes generated by the Branch & Bound method during the solution of problem (3). Then X does not

contain a feasible solution of problem (3) if and only if L is not a feasible solution of problem (3), namely $C_{nI_n} > D$.

Theorem 4. Suppose that $X = [L, U]$ is one of the sub-boxes generated by the Branch & Bound method during the solution of problem (3). For all $x \in X \cap I^n$, if $x \neq U$, then $f(x) > f(U)$. And if U is a feasible solution of problem (3), then U is an optimal solution of following sub-problem.

$$\begin{cases} \min & f(x) = \sum_{i=1}^n b_{ix_i} \\ \text{s.t.} & C_{nx_n} \leq D \quad x = (x_1, x_2, \dots, x_n)^T \in X \cap I^n \end{cases} \tag{4}$$

Theorem 5. Suppose that $X_1 = [L^1, U^1]$ and $X_2 = [L^2, U^2]$ are two sub-boxes generated by the Branch & Bound method during the solution of problem (3).

If L^2 is a feasible solution and $L^2 > U^1$, then there exists $x \in X_1 \cap I^n$, such that $f(x) \geq f(U^1) > f(L^2)$, ie, X_1 does not contain an optimal solution of problem (3).

For a box $X = [L, U]$ generated by the partition method. If L is a feasible solution of problem (3) but U is not. Then we can retrench the box X using the following method. Let

$$A_i = \max\{a : L + ae_i \in X \cap I^n, \text{ and } L + ae_i \text{ is a feasible solution}, i = 1, 2, \dots, n\} \tag{5}$$

$$U_i = l_i + a_i, i = 1, 2, \dots, n, U = (u_1, u_2, \dots, u_n)^T \tag{6}$$

Where e_i is a vector in witch the i th component equals to 1 and the other components equal to 0.

Since C_{nx_n} is a monotonically increasing function, we use the binary search method to solve Eq. (5).

Our computational experiences show that retrenches a box X using this method can improve the computational speed of the Branch & Bound method.

The basic idea for the solution of problem (3) by the Branch & Bound method is described as follows.

Step 1: $Q := \{[E, M]\}, f^* := +\infty$.

Step 2: Let X be the box which have the maximal edge length in Q . Let $Q := Q \setminus \{X\}$. Bisect X into $X_1 = [L^1, U^1]$ and $X_2 = [L^2, U^2]$ using the division method mentioned above.

Step 3: For $X_j, j = 1, 2,$

Step 3.1: If L^j is not a feasible solution of problem (3) then delete X_j .

Step 3.2: Calculate $a_i^j = \max\{a : L^j + ae_i \in X \cap I^n, \text{ and } L^j + ae_i \text{ is a feasible solution}\}, i = 1, 2, \dots, n$ and retrench boxes X_1 and X_2 according to (5), (6). The retrenched boxes are also written as $X_1 = [L^1, U^1]$ and $X_2 = [L^2, U^2]$

Step 3.3: Let $f^* := \min\{f^*, f(L^j + a^j e_i), i = 1, 2, \dots, n\}$,

Step 3.4: If U^j is a feasible solution of problem (3), then delete X_j , and if $f(U^j) < f^*$, then let $f^* := f(U^j), x^* := U^j$.

Step 3.5: If $f(U^j) \geq f^*$, then delete X_j .

Step 4: Enter the undeleted sub-boxes in Step 3 into Q .

Step 5: If Q is not empty. Then go to Step 2. Otherwise output f^* and x^* as the minimal value and minimal solution of problem (3).

It is easy to show that The Branch & Bound method has the following property:

Theorem 6. The Branch & Bound method can find an optimal solution of problem (3) or conclude that problem (3) has no feasible solutions.

4 Numerical Experiments

In this part, we generate randomly some examples of the resource selection problem and test the performance of the Branch & Bound method and contract to the bid algorithm in literature^[6]. The dimensions of testing examples are 10, 20, 30, 40, 50 and 100 respectively. The number of candidates for each subtask is 10. After the definition of n and m . An activity network is generated randomly. For each subtask of the activity network, the completion time and cost of each candidate is generated randomly. For each pair of n and m , we generate randomly 10 testing examples, and average the computational time of The Branch & Bound method. The test results are time 1, and the test results are time 2 based on the bid invitation and bidding algorithm in literature, they are put in Table 1.

Table 1. The test result contrast of the Branch & Bound method and the invitation and bidding algorithm

Problems	n	m	Time1(s)	Time2(s)
1-10	10	10	1.83	3.35
11-20	20	10	12.65	150.23
21-30	30	10	34.63	626.37
31-40	40	10	265.7	4893.03
41-50	50	10	927.6	-----

From the contrast result, it can be seen that a Branch & Bound method increases more superiority in the determination resources aspect along with the question scale increasing. When the magnitude of the problem increases to a certain extent, the search speed of this algorithm will be greatly reduced and even no optimal solution can be found, it is because among the resources Agent frequent data transmission, but a Branch & Bound method can find the optimal solution in the limited time.

5 Conclusions

We have proved that the resource selection problem with a due date and precedence constraint is NP-complete problem, and established a nonlinear integer programming model for this problem. We have also constructed a Branch & Bound method to solve the problem. Numerical experiments show that the algorithm is efficient.

References

1. Liu, H.L., Wu, T.J.: Distributed Algorithm for Task Allocation in Multi-agent System Based on Contract Net. *Journal of Zhejiang University (Engineering Science)* 35, 550–554 (2001)
2. Xiong, R., Wu, C.: Current Status and Developing Trend of Job Shop Scheduling Research. *Journal of Tsinghua University (Science and Technology)* 38, 55–60 (1998)
3. Talluri, S., Baker, R.C.: Quantitative Framework for Designing Efficient Business Process Alliance. In: *Proceedings of 1996 International Conference on Engineering and Technology Management*, Piscataway, pp. 656–661 (1996)
4. Elmaghraby, S.E.: *Activity Networks-task Planning and Control by Network Models*. Wiley, New York (1977)
5. Wang, D., Yung, K.L., Ip, W.H.: A Heuristic Genetic Algorithm for Subcontractor Selection in a Global Manufacturing Environment. *IEEE Transactions on Systems, Man and Cybernetics – Part C: Applications and Reviews* 31, 189–198 (2001)
6. Li, Y., Li, S.J., Liu, Y.: Dynamic Scheduling Method Based on Combination of Contract Net with Mediator. In: *Proceedings of 2005 international conference on machine learning and cybernetics*, Guangzhou, China, vol. 1, pp. 339–344 (2005)
7. Shehory, O., Kraus, S.: Methods for Task Allocation via Agent Coalition Formation. *Artificial Intelligence*, vol. 101, pp. 165–200 (1998)
8. Garey, M.R., Johnson, D.S.: *Computers and Intractability: A Guide to the Theory of NP-completeness*. WH Freeman and Company, New York (1979)

Rule-Based Modeling of Assembly Constraints for Line Balancing

Latif Salum and Aliye Ayca Supciller

Dokuz Eylul University, Muhendislik Fak., End. Bol., Bornova, Izmir, 35100, Turkey
{latif.salum, ayca.supciller}@deu.edu.tr

Abstract. The assembly line balancing problem employs traditional precedence diagrams to model precedence relations among assembly tasks. Yet they cannot address alternative ways of assembling a product. That is, they only model conjunctions (AND relations), not disjunctions (OR relations). However, there may be some alternative assembly plans for a product. Moreover, some complicated constraints need also to be considered, but these constraints cannot be modeled effectively through traditional precedence diagrams. This paper proposes a rule-based representation to address these issues. The rule-base alone can model any constraint effectively. The paper also shows that the rule-base can easily be employed by heuristics used for line balancing problems.

Keywords: Assembly line balancing; Precedence constraints; Rule-based representation.

1 Introduction

The assembly line balancing (ALB) problem is the decision problem of optimally partitioning (balancing) the assembly work among workstations [1]. Any ALB problem consists of at least three basic elements: a precedence graph (diagram) which comprises all tasks and resources to be assigned, the stations which make up the line and to which those tasks are assigned, and some kind of objective to be optimized [2]. The authors classify the ALB problem based on these three elements.

There are some shortcomings of the precedence diagrams. They usually fail to represent all the possible assembly sequences of a product in a single diagram [3], and exclude some logic statements, e.g., the precedence relation “(2 or 3) → 7” cannot be represented properly on a precedence diagram [4]. Hence, they allow a limited flexibility. Alternative precedence sub-graphs may be needed when there are assembly alternatives, and the system designer normally selects a priori one alternative to determine the precedence graph [5]. Precedence diagrams fail to describe some complicated constraints, e.g., constraints indicating that some pairs of tasks cannot be assigned into the same station because of incompatibility between them caused by some technological factors [6].

Yet despite their shortcomings, researchers continue to employ precedence diagrams without questioning [7]. There are also some alternative representation methods, e.g., AND/OR graphs [8], used in the line balancing problem [7]. Koç [7] discussed the applicability of precedence diagrams and showed that AND/OR graphs were better than precedence diagrams for the line balancing problem. Capacho and

Pastor [5] employed some alternative assembly sub-graphs, in which processing times and/or precedence relations of certain tasks may vary, and solved the ALB problem by simultaneously selecting an assembly sub-graph and balancing the line. Park et al. [6] introduced two sub-problems to further consider some incompatibility constraints, range constraints, and partial precedence constraints.

This paper proposes a rule-based representation for assembly constraints that overcomes the aforementioned difficulties. This representation is more effective compared to the approaches above, as all the (assembly) constraints can be modeled conveniently in a rule-base alone, and this rule-base can then easily be employed by any assembly line balancing heuristics.

2 Rule-Based Representation of Constraints

The rule-based modeling is demonstrated through an example. The assembly problem is derived from apparel industry, sewing of a simple pant. The task times are given in Table 1. The cycle time is assumed to be 2 minutes. The sum of the task times is 7 minutes. Thus, at least four workstations are necessary ($7 / 2 = 3.5$).

The precedence relations are also shown in Table 1. For example, task 1, T_1 , can be assigned to any workstation without any precedence constraint. T_5 can be assigned

Table 1. Tasks of a simple pant assembly

Precedence Relation	Task T_i	Time (min)	Description
—	1	0.40	overlock stitch of parts of front right pocket
—	2	0.35	overlock stitch of parts of front left pocket
1	3	0.75	overlock stitch of front right pocket and front right part
2	4	0.80	overlock stitch of front left pocket and front left part
(9, 10) OR (3, 4)	5	0.60	overlock stitch of front left part and front right part
— OR (9, 10)	6	0.55	overlock stitch of back left part and back right part
5, 6	7	0.50	inside overlock stitch of back left part and back right part
7	8	0.45	inside overlock stitch of front left part and back left part
3 OR 8	9	0.70	outside overlock stitch of back left part and back right part
4 OR 9	10	0.60	outside overlock stitch of front left part and back left part
8 OR 10	11	0.80	stitch of waist
11	12	0.50	stitch of leg opening

after T_3 and T_4 , or T_9 and T_{10} are assigned. T_6 can be assigned without any precedence constraint or after T_9 and T_{10} are assigned. Note that task 5, 6, 9, 10 and 11 has some alternative assembly precedence relations, which cannot be modeled through conventional precedence diagrams easily. They can be modeled through some alternative assembly sub-graphs, but it is difficult to derive them from such a table, which is the basic input and most commonly used tool in ALB problems. Moreover, there are some complicated constraints that cannot be easily modeled by assembly sub-graphs, which are basically modified precedence diagrams.

These precedence constraints can be modeled through a knowledge base of If-Then rules more conveniently. The precedence constraints can also be modeled through Colored AND/OR Petri Nets [9]. As a graphical tool, the CARPN model is used for validation, and communication among users and designers. As a mathematical tool, it is used for verification of the assembly through place invariants analysis: if all places are covered by the place invariants, then the assembly is verified, i.e., all the subassemblies and components can be put together in the final product assembly. Transitions *enabled* and *fired* correspond to tasks *assignable* and *assigned*, respectively.

The If-then rules can easily be derived from Table 1; the precedence relation of each task is simply mapped to an If-then rule. The rules are defined below.

- R1: If T_1 then T_3
- R2: If T_2 then T_4
- R3: If $(T_9 \text{ AND } T_{10}) \text{ OR } (T_3 \text{ AND } T_4)$ then T_5
- R4: If $T_6 \text{ OR } (T_9 \text{ AND } T_{10})$ then T_6
- R5: If $T_5 \text{ AND } T_6$ then T_7
- R6: If T_7 then T_8
- R7: If $T_3 \text{ OR } T_8$ then T_9
- R8: If $T_4 \text{ OR } T_9$ then T_{10}
- R9: If $T_8 \text{ OR } T_{10}$ then T_{11}
- R10: If T_{11} then T_{12}

There are as many rules as tasks with some precedence relations. As there are 12 tasks, and T_1 and T_2 have no precedence relations, i.e., they are assignable initially, the number of the rules is 10.

Some fuzzy rules can also easily be employed in such a rule-base to model vagueness in assembly constraints. Recall that some complicated constraints can also be modeled easily through a rule-base. For example, if certain task times vary with respect to assembly sub-graphs, e.g., see Capacho and Pastor [5], a rule of the form "if $T_x \rightarrow T_y$ then $P(T_x) = 5$ " is used to mean that the task time of T_x is five minutes if T_x precedes T_y . Similarly, if a constraint indicates that certain tasks cannot be assigned into the same station, e.g., see Park et al. [6], a rule of the form "if $T_x \in S_k$ then $T_y \notin S_k$ OR if $T_y \in S_k$ then $T_x \notin S_k$ " is used to mean that tasks T_x and T_y cannot be assigned into the same station, S_k .

The designer then uses the rule-base above, instead of a precedence diagram, and any heuristic for the assembly line balancing. This paper exploits a simple heuristic for this simple problem, the largest candidate rule, e.g., see Groover [10], which gives top assignment priority to the task with the longest task time to determine tasks to be

assigned into stations. Some other heuristics like genetic algorithms (GAs) can also be employed, as discussed later.

Initially, T_1 and T_2 are assignable, i.e., they are evaluated to be “true”. T_6 is also assignable initially, due to R4. Note that a rule fires as soon as it is enabled, without awaiting all the variables in its antecedent to be “true”, i.e., all the relevant tasks to be assigned. Hence, R4 fires without awaiting T_9 and T_{10} to be also “true”. If R4 firing awaited all the antecedent variables to be “true” (if possible), T_9 and T_{10} would also be considered in this assignment. The solution qualities of these two policies should be compared through several experiments.

Consequently, T_1 , T_2 and T_6 are assignable at the outset. Because the task time of T_6 is the longest, i.e., it is at the top of the list, T_6 is assigned to the first station first. The tasks assignable are then T_1 and T_2 . Since T_1 is at the top of the list, it is assigned to the first station next. R1 then fires, which makes T_3 assignable, besides T_2 . Since T_3 is at the top of the list, it is assigned to the first station next. R7 then fires, which makes T_9 assignable. The tasks assignable are then T_2 and T_9 . Because the idle time of the first station is 0.30, none of them can be assigned to the first station, and considered for the second station. T_9 is assigned to the second station since it is at the top of the list. R8 then fires, and the assignable tasks are T_2 and T_{10} . All the tasks are assigned in this manner, and Table 2 is obtained. *11^(R9) in Table 2 means that although T_{11} is assignable due to R9, it cannot be considered for the second station as its task time, 0.80, exceeds the station idle time, 0.70. Note that this process corresponds to the “forward chaining” in expert systems, and yields the firing sequence of R4, R1, R7, R8, (R3, R9), R5, R10, R6, and R2.

The first station includes T_6 , T_1 , and T_3 ; the second station T_9 , T_{10} , and T_5 ; the third station T_{11} , T_7 , and T_{12} ; and the fourth station T_8 , T_2 , and T_4 . Since the total task time is 7 minutes, the balance efficiency is $92\% = 7 / 4 \times 1.90$. Note that the italic idle times indicate the actual idle times, as the actual cycle time is 1.90 min.

As mentioned, GAs are commonly used heuristics in the line balancing problem, e.g., see Scholl and Becker [11]. If necessary, the penalty term for precedence

Table 2. Assignment of the tasks to stations subject to the rule-base

Station	Tasks assignable	Task assigned	Station idle time	Station time
1	1, 2, 6 ^(due to R4)	6 ^(R4 fires)	1.45	0.55
	1, 2	1 ^(R1 fires)	1.05	0.95
	2, 3 ^(due to R1)	3 ^(R7 fires)	0.30 – 0.20	1.70
2	2, 9 ^(R7)	9 ^(R8)	1.30	0.70
	2, 10 ^(R8)	10 ^(R3, R9)	0.70	1.30
	2, 5 ^(R3) , *11 ^(R9)	5 ^(R5)	0.10 – 0.0	1.90
3	2, 7 ^(R5) , 11 ^(R9)	11 ^(R10)	1.20	0.80
	2, 7 ^(R5) , 12 ^(R10)	7 ^(R6) , 12	0.20 – 0.10	1.80
4	2, 8 ^(R6)	8	1.55	0.45
	2	2 ^(R2)	1.20	0.80
	4 ^(R2)	4	0.40 – 0.30	1.60

violations in the fitness function can be calculated easily through evaluation of every rule. For example, if a chromosome decodes that the second station includes T_2 , T_4 , and T_7 ; and the third T_5 , T_8 , and T_9 , then the number of the violated precedence constraints (rules) is one, due to R5. That is, because T_6 and T_5 should be completed before T_7 , T_7 in the second station and T_5 in the third violate R5.

3 Rule-Based Representation vs. Precedence Graphs

One can suggest that traditional precedence graphs can also be derived from Table 1, although it is very difficult to derive them if the precedence relations are more complex. Figure 1 is the graph-based representation of Table 1. However, it is not the exact representation, i.e., Figure 1 is not equivalent to the rule base, because the two graphs are mutually exclusive, but the rule base is not. Their performances are

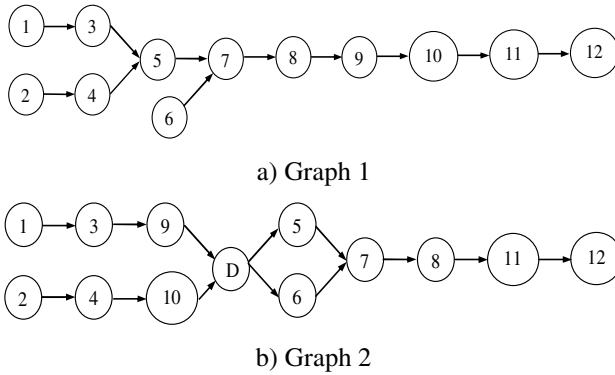


Fig. 1. Representation of Table 1 with two precedence diagrams

Table 3. Assignment of the tasks to stations subject to Figure 1a

Station	Tasks assignable	Task assigned	Station idle time	Station time
1	1, 2, 6	6	1.45	0.55
	1, 2	1	1.05	0.95
	2, 3	3	0.30 – 0.20	1.70
2	2	2	1.65	0.35
	4	4	0.85	1.15
	5	5	0.25 – 0.15	1.75
3	7	7	1.50	0.50
	8	8	1.05	0.95
	9	9	0.35 – 0.25	1.65
4	10	10	1.40	0.60
	11	11	0.60	0.40
	12	12	0.10 – 0.0	1.90

compared below under the mutual exclusion assumption. The reader can refer to [12] for more details.

If Figure 1a is considered in the line balancing, Table 3 is obtained. Table 3 yields the same balance efficiency as the rule base, 92%.

If Figure 1b is considered in the line balancing, Table 4 is obtained. The balance efficiency is $95\% = 7 / 4 \times 1.85$.

Table 4. Assignment of the tasks to stations subject to Figure 1b

Station	Tasks assignable	Task assigned	Station idle time	Station time
1	1, 2	1	1.60	0.40
	2, 3	3	0.85	1.15
	2, 9	9	0.15 – 0.0	1.85
2	2	2	1.65	0.35
	4	4	0.85	1.15
	10	10	0.25 – 0.10	1.75
3	5, 6	5	1.40	0.60
	6	6	0.85	1.15
	7	7	0.35 – 0.20	1.65
4	8	8	1.55	0.45
	11	11	0.75	1.25
	12	12	0.25 – 0.10	1.75

Based on the results above, Graph 2 in Figure 1b outperforms the others. However, this is due to the line balancing heuristic, rather than the inefficiency of the rule-base. In fact, the optimum solution can be found for these problems [12]. The optimum balance efficiency is 100%, 92%, and 95% for the rule-base, Figure 1a, and Figure 1b, respectively. The 100% efficiency in the rule-base is obtained with the following assignment: (T_2, T_4, T_{10}) , (T_1, T_3, T_5) , (T_6, T_7, T_9) and (T_8, T_{11}, T_{12}) , which yields the cycle time of 1.75 min. Note that due to the inclusiveness of the rule-base, which does not hold between Graph 1 and Graph 2 in Figure 1, it is possible to assign (T_2, T_4, T_{10}) in Graph 2 to the first station, and (T_1, T_3, T_5) in Graph 1 to the second station. That is, the inclusiveness property of the rule-base makes it possible to combine various tasks in the two graphs, which improves the efficiency.

4 Conclusion

The major drawback of precedence diagrams is that they are not suitable to model alternative ways of assembling a product. Moreover, some complicated constraints need also to be considered, but these constraints cannot be modeled effectively through traditional precedence diagrams. This paper proposed a rule-based representation to tackle these issues for assembly line balancing problems. All the assembly alternatives and other constraints can be represented through a rule-base alone, a knowledge base of If-Then rules. Any line balancing heuristic then uses this

rule-base. The line balancing process corresponds to the “forward chaining” in expert systems. Some fuzzy rules can easily be employed in such a rule-base to model vagueness in assembly constraints. Instead of the rule-base, Colored AND/OR Petri Nets can also be used, which enables verification of the assembly through place invariants analysis. The focus of the paper was on modeling the assembly constraints rather than to propose a line balancing heuristic. A simple example was given to demonstrate how this representation could be used in assembly line balancing. The future work will focus on GA-based heuristics and constraint programming formulations based on a rule-based model, and the complex-constrained assembly line balancing problem will be introduced [12].

References

1. Scholl, A.: *Balancing and Sequencing of Assembly lines*, 2nd edn. Springer, New York (1999)
2. Boysen, N., Fliedner, M., Scholl, A.: A Classification of Assembly Line Balancing Problems. *European Journal of Operational Research* 183, 674–693 (2007)
3. Lambert, A.J.D.: Generation of Assembly Graphs by Systematic Analysis of Assembly Structures. *European Journal of Operational Research* 168, 932–951 (2006)
4. De, F.T.L., Whitney, D.E.: Simplified Generation of All Mechanical Assembly Sequences. *IEEE Journal of Robotics and Automation* 3, 640–658 (1987)
5. Capacho, L., Pastor, R.: The ASALB Problem with Processing Alternatives Involving Different Tasks: Definition, Formalization and Resolution. In: Gavrilova, M.L., Gervasi, O., Kumar, V., Tan, C.J.K., Taniar, D., Laganá, A., Mun, Y., Choo, H. (eds.) ICCSA 2006. LNCS, vol. 3982, pp. 554–563. Springer, Heidelberg (2006)
6. Park, K., Park, S., Kim, W.: A Heuristic for an Assembly Line Balancing Problem with Incompatibility, range, and Partial Precedence Constraints. *Computers and Industrial Engineering* 32, 321–332 (1997)
7. Koç, A.: Inquiring the Main Assumption of the Assembly Line Balancing Problem: Solution Procedures Using AND/OR Graph. MSc Dissertation, Bilkent University, Ankara (2005)
8. Homem, M.L.S., Sanderson, A.C.: AND/OR Graph Representation of Assembly Plans. *IEEE Transactions on Robotics and Automation* 6, 188–199 (1990)
9. Salum, L.: A New Class of a High-level Petri net for Modelling Logical OR Efficiently: Coloured AND/OR Petri Nets (CARPN). *International Journal of Production Research* 38, 4671–4682 (2000)
10. Groover, M.P.: *Automation, Production Systems, and Computer-integrated Manufacturing*, 2nd edn. Prentice Hall, New Jersey (2001)
11. Scholl, A., Becker, C.: State-of-the-art Exact and Heuristic Solution Procedures for Simple Assembly Line Balancing. *European Journal of Operational Research* 168, 666–693 (2006)
12. Salum, L., Topaloglu, S., Supciller, A.A.: The Complex-constrained Assembly Line Balancing Problem. *European Journal of Operational Research* (under review, 2008)

A Novel Network Behavior Model Based on Generalized Cellular Automaton

Dianxun Shuai

East China University of Science and Technology, Shanghai 200237, China
shdx411022@online.sh.cn

Abstract. Computer networks exhibit very complex behavior.¹ To establish an effective model for network behavior is of significance for improving the network performance. This paper presents a new network behavior model based on generalized cellular automaton. The proposed model may take into consideration the different personality and autonomy of network entities, the social interactions among entities, and the influences of concurrent emergent events in networks.

Keywords: generalized cellular automaton; dynamical behavior; computer networks; agent colonies; macroscopic performance.

1 Introduction

With the explosive growth of the Internet, the massive information networks have become the hyper-distributed hyper-parallel open giant-complex system, and exhibit the complicated social interactive behaviors, the quasi-organic metabolic behaviors and the non-linear dynamic behaviors. The research on network behaviors is of great significance not only for modeling network dynamics, but also for optimizing resource assignment, controlling traffics and congestions, ensuring network security, and improving information access and its exploitation. In recent years, several efforts have been devoted to modeling the network traffic and the cyclical evolution of the TCP transmission window. Lakshman et al.^[1–3] used Markovian analysis to develop a closed-form expression for the throughput of TCP connection. Mathis et al.^[4] focused on stochastic behavior of the congestion avoidance mechanism, deriving an expression for the throughput. Padhye et al.^[5] have developed a steady-state model that approximates the throughput of bulk TCP flows as a function of loss rate and round trip time. Notably, it has been discovered so far that, no matter how the network topology, the number of users and the network service types changed, a self-similarity fractal model is much better suitable for representing network traffic in comparison with the classical Markov model^[6–10]. Moreover, the social mind and social behaviors the distinct individuals in the networks try to take will greatly affect on the network performance and network behaviors. Huberman and Lukose classified the Internet users into two kinds: cooperators and defectors, the former restrain their

¹ This work was supported by the National Natural Science Foundation of China under Grant No.60575040 and No.60473044.

use of the network and the later consume the network bandwidth greedily. In fact, because Internet surfers are not charged in proportion to their use, it often appears rational for individuals to consume bandwidth greedily while thinking that their actions have little effect on the overall performance of the Internet. Because every individual can reason this way, the whole Internet's performance can degrade considerably, which makes everyone worse off. Internet is not immune to the problems intrinsic to any public good, as congestions sometimes threaten to render it useless. But once users experience a congested network, they will reduce or even desist in their use, freeing up bandwidth that can once again be consumed, and so on. Mimicking this social dilemma by a differential equations, Huberman et al. demonstrated that intermittent sudden spikes of congestion occur and possess non-Gaussian statistical properties.

We guess that, like other open dissipative non-linear system, the Internet must generate some self-organizing temporal-spacial behavior patterns and some phase-transitive phenomena related to the key ordering parameters. In this paper, we present a new generalized cellular automaton (GCA) approach to model dynamic behavior of computer networks. The proposed approach is featured by the feasibility to deal with complicated social interactions among network agents, and the ability to combine the autonomy of individual agent with the intelligence of agent colonies of various granularity. The simulations on the network dynamic behaviors are given, which show the flexibility and effectiveness of our methodology.

2 Architecture and Dynamics of GCA

A pyramid hierarchical conceptual architecture for modelling the dynamics of networks is sketched in Fig.1, which is composed of several layers, each layer is an array of macro-cells with the roughly equal cell-granularity. The higher a layer is located, the larger granularity the macro-cells in the layer have. Moreover, a macro-cell in a given layer consists of several macro-cells in the lower layer and becomes a constituent of some macro-cells in the higher layer, which is essentially different from general hierarchy where the members in distinct layers are almost entirely independent individuals without inclusive relationship between them. The macro-cells in the lowest layer are primitive cells, whereas the macro-cell in the highest layer can be regarded as the whole system under consideration. And hence the number of macro-cells in a layer gradually decrease from bottom layer to top layer. A macro-cell will be affected by the social interactive behaviors of other neighbouring macro-cells in the same layer, and will receive the performance feedback from a relative macro-cell in the higher layer. Furthermore, a stratified complex composed of many pyramid hierarchical architectures shown in Fig.1 can be constructed to describe and to model much more complicated dynamic behaviors of networks, as shown in Fig.2, each stratification being considered as a macro-cell with huge cell-granularity and interactions between different stratifications being able to occur.

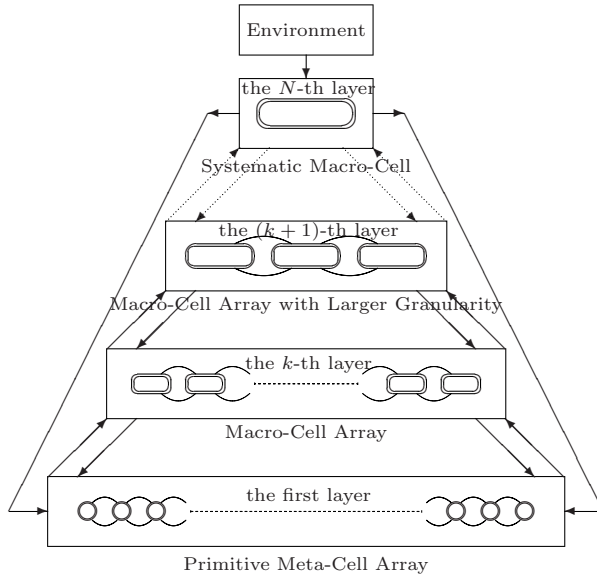


Fig. 1. A pyramid hierarchical conceptual architecture of generalized cellular automaton

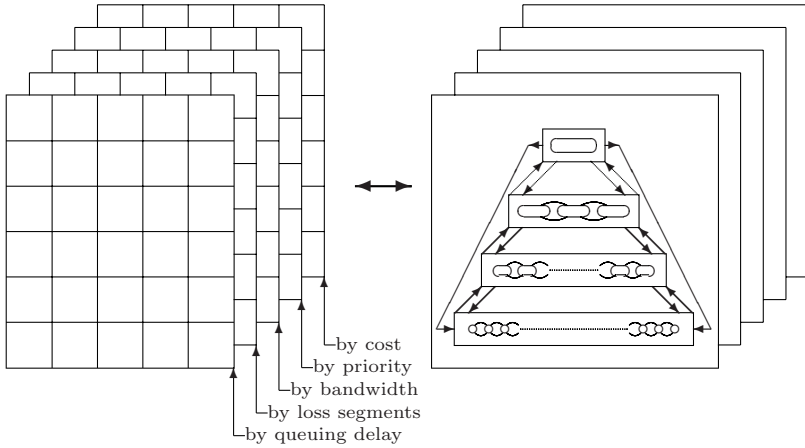


Fig. 2. The stratified complex composed of several pyramid hierarchical architectures of the GCA (Each pyramid generalized cellular automaton corresponding to some specific network behaviours)

As for a macro-cell array in the pyramid hierarchical architecture of the GCA, like the CNN (Cellular Neural Networks), its constitutional macro-cells also have two basic features: The local communications with each other directly occur only through their designated neighbors; And the large-scale nonlinear continuous

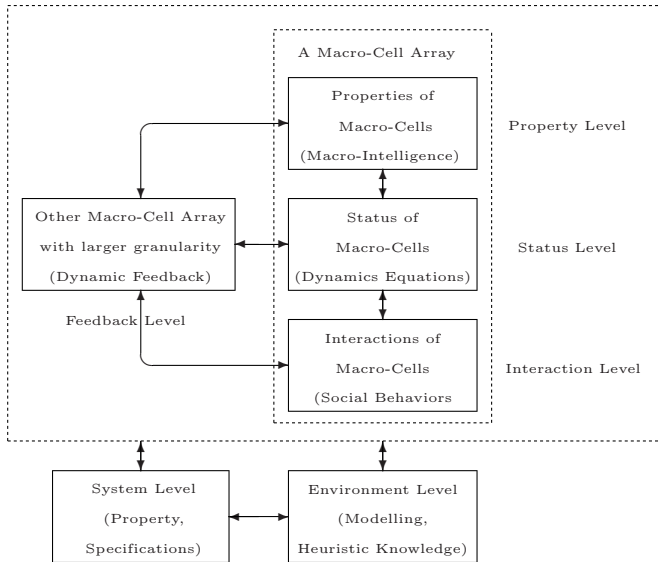


Fig. 3. The dynamic levels relative to a macro-cellular array within GCA

analog signal dynamics are defined by a set of differential equations. On the other hand, however, our paradigm is essentially different from the CNN in the following aspects: The strengths and directions of local connections between macro-cells in a layer are entirely determined by the social interactive behaviors between them, and are thus dynamically changeable and not previously fixed; The dynamic differential equation of a micro-cell in a layer contains the terms relative to performance feedback from the macro-cells in a higher layer with larger cell-granularity, which implies a distributed parallel closed-loop model. Moreover, in our model, the dynamics of a macro-cell array is referred to the following six levels, as given in Fig.3: The interaction level is concerned with the social interactive behavior space made up of various concurrent stochastic complicated social coordinations among the macro-cells belonging to the same macro-cell array; The status level is described by a set of non-linear dynamic differential equations for every macro-cell in the array and represents a set of possible states every macro-cell in the array might have; The property level means the space of the macroscopic feature or intelligence that every macro-cell in the array exhibits; The feedback level exerts an impact on a macro-cell array through the performance feedback stemmed from higher macro-cell array which has larger cell-granularity; The system level specifies the necessary specification and evaluates the global performance; Finally, the environment level provides the GCA with the initial conditions, parameters and some heuristic knowledge if necessary.

Consider a matrix $M = [M_{ij}] \in R^{n \times m}$ which can formally express some network problems concerning planning, controlling or modelling, as shown in

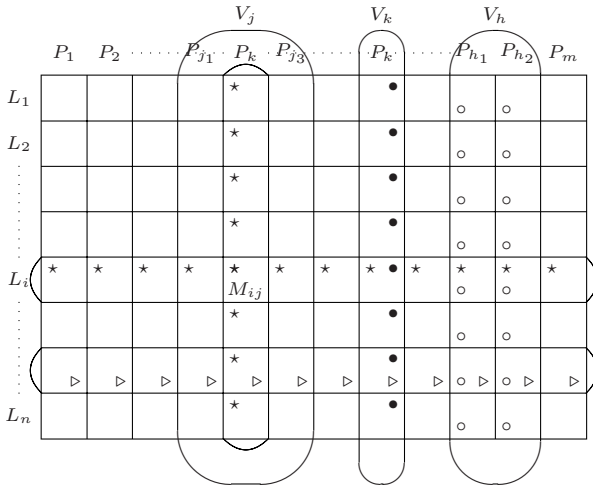


Fig. 4. The transformation from a formalization matrix $M = [M_{ij}] \in R^{n \times m}$ of network problem into a generalized macro-cellular automaton

Fig.4. Each $M_{ij}, 1 \leq i \leq n, 1 \leq j \leq m$, is regarded a primitive macro-cell with the smallest granularity, and its neighbour is composed of all the macro-cells that are located in the i -th row or the j -th column of the M , which results in that the lowest layer of the pyramid hierarchic architecture has $n \times m$ macro-cells. We can define a macro-cell with larger granularity than M_{ij} , and particularly, define a row $M_i = [M_{ij}] \in R^{1 \times m}$ or a column $M_j = [M_{ij}] \in R^{n \times 1}$ as a macro-cell, whose neighbor consists of all the rows or columns of the M , respectively. Furthermore, in some cases (e.g., network bandwidth assignment dynamics), several rows or columns of the M should be seen as a macro-cell. Extremely, the whole M is also considered as a macro-cell, whose neighbor maybe involves other two-dimensional matrices, as illustrated in Fig.2. Every macro-cell in a layer, no matter whether its granularity is small or large, has its own dynamic differential equation which synthesizes its subjective autonomous behavior, the social interactive behavior from other macro-cells in the same layer and the adaptive behavior from performance feedback of other macro-cells in a higher layer with larger macro-cellular granularity.

Hereafter, we use n rows of a matrix M to represent a set $E = \{E_1, \dots, E_n\}$ of network entities, and m columns a set $R = \{R_1, \dots, R_m\}$ of network requests. Each network entity E_i has an ability vector $\bar{e}_i = (e_i^1, \dots, e_i^{k_i})$ and gives a pricing vector $\bar{c}_i = (c_i^1, \dots, c_i^{k_i})$ for unit ability vector, and each network request R_j requires an ability vector $\bar{d}_j = (d_j^1, \dots, d_j^{k_j})$ and provides a paying vector $\bar{p}_j = (p_j^1, \dots, p_j^{k_j})$ for unit ability vector. The entry \bar{a}_{ij} of the $M, i \in \{1, \dots, n\}, j \in \{1, \dots, m\}$, is time-varying problem-dependent vector, such as assigned ability vector, queuing delay, packet loss, throughput, traffic parameter and so on.

For the i -th macro-cell $m_i^{(k)}$ in the k -th macro-cellular layer of Fig.1, its dynamic differential equation is defined by

$$\begin{cases} dx_i^{(k)} / dt = T_1(t) + T_2(t) \\ T_1(t) = -x_i^{(k)}(t) + \alpha_i y_i^{(k)}(t) \\ T_2(t) = \sum_{j \in \mathcal{N}(i,t), j \neq i} \sigma_{ij}^{(k)}(t) y_j^{(k)}(t) + g_i^{(k)}(t) + f_i^{(k)}(t) + x_i^{(k)}(t_0) + I_i \\ y_i^{(k)}(t) = \varphi(x_i^{(k)}(t)) \end{cases} \quad (1)$$

where $x_i^{(k)}(t)$ is the output of the i -th macro-cell at time t ; $y_i^{(k)}(t)$ is its inner state at time t ; $g_i^{(k)}(t)$ is the feedback function stemmed from related macro-cells in the $(k + 1)$ -th macro-cellular layer; $f_i^{(k)}(t)$ represents the constraint condition concerning $x_i^{(k)}(t)$ at time t ; I_i is a bias constant; $\alpha_i > 1, \sigma_{ij}(t_0) \in R$; $\mathcal{N}(i, t)$ is the neighbor of the i -th macro-cell at time t , of which every macro-cell is also located in the k -th layer and has some social interactions with the considered i -th macro-

cell; $\varphi(x(t))$ is a piecewise linear function, $\varphi(x(t)) = \begin{cases} 0 & \text{if } x(t) < 0 \\ x(t) & \text{if } 0 \leq x(t) \leq 1 \\ 1 & \text{if } x(t) > 1, \end{cases}$ namely, $\varphi(x(t)) = [|x(t)| - |x(t) - 1| + 1]/2$.

Remarks 1:

- The first term $\sum_{j \in \mathcal{N}(i,t), j \neq i} \sigma_{ij}(t_0) y_j(t)$ of $T_2(t)$ expresses the effect caused by the interactions between the i -th macro-cell and other macro-cells in its neighbor $\mathcal{N}(i, t)$, whose coefficient $\sigma_{ij}(t)$ describes the type and the strength of interaction between the i -th and j -th macro-cells.

- In general, $g_i(t) =_{DF} \frac{\partial x_j^{(k+1)}}{\partial x_i^{(k)}} \Big|_t$ is used to express the effect generated for the i -th macro-cell $m_i^{(k)}$ by macroscopic performance of a macro-cell $m_j^{(k+1)}$ which has component $m_i^{(k)}$, where $x_j^{(k+1)}$ may be a Lyapanov function, energy function or error function. If there is no $T_1(t)$ and there not exist other terms in $T_2(t)$ except for the term $g_i^{(k)}(t)$, the dynamics (1) is simplified to an usual equation of gradient method.

- The $T_1(t)$ embodies such an autonomy of the i -th macro-cell as tries to increase its own outcome.

3 Simulations and Conclusions

Example 1. The network dynamics associated with the marketing mechanism.

In what follows, we outline the network dynamics of bandwidth assignment associated with the marketing mechanism. Given a set $V = \{(v_i, v_j)\}$ of network node pairs (v_i, v_j) between which communication with bandwidth b_{ij} is requested, each node pair (v_i, v_j) has the corresponding set $P_{ij} = \{P_{ij}^1, \dots, P_{ij}^{k_{ij}}\}$ of possible virtual paths, and each $P_{ij}^k, k \in \{1, \dots, k_{ij}\}$ requires a certain bandwidth and is a semi-persistent connection formed by the cascade of L_{ij}^k links. Several virtual paths maybe need the same link and share the bandwidth of the

link simultaneously. A link has its allowable maximal bandwidth. Corresponding to Fig.4, we regard a link as a network entity, a virtual path as a network request, and the entry \bar{a}_{ij} of the M as the assigned bandwidth. As for the structure of a macro-cell, there must be such a macro-cell that contains all the columns corresponding to virtual paths via which a pair of network nodes possibly uses to communicate at the same time. Fig.4 demonstrates three macro-cells, V_j, V_k and V_h , which include paths $\{p_{j1}, p_{j2}, p_{j3}\}, \{p_k\}$ and $\{p_{h1}, p_{h2}\}$ respectively.

Suppose that each link tries to increase its own benefit through adjusting its pricing policy, and, meanwhile, each virtual path manages to be charged as less as possible via changing its paying policy under the condition that its demand is satisfied. Therefore, if a link is shared by many virtual paths, the link will raise its pricing for unit bandwidth and vice versa. Similarly, if a bandwidth request of a virtual path acquires exceeding responses from links so that the bandwidth provision from them greatly exceeds the demand of the path, then the path will decrease its paying for unit bandwidth, and vice versa. In this way, the bandwidth assignment process exhibits evolutionary dynamics brought about by marketing mechanism.

Example 2. The network dynamics caused by social interactions.

Next, we discuss such network dynamics that a number of network agents try to take over a set of network tasks through various social coordinations. Each network agent wants to participate in more profitable task through interactions with other agents, whereas each network task is willing to be taken up by more powerful agents through interactions with other tasks. The network agents and network tasks correspond to network entities and network requests in Fig.4, respectively. The social interactions are referred not only to competition, cooperation and coalition, but also to other unilateral social coordinations, such as deception, induction, exploitation, etc. Thus the task allocation dynamics will be governed by the strength and character of social interactive behaviors.

Example 3. The network dynamics associated with users' mental or users' adaptive behavior.

Given a network configuration (e.g., routing, scheduling, buffer management), the network users represent the network requests in Fig.4, who consume bandwidth often greedily while thinking that their actions have little effect on the overall performance of the network. But as users experience an intolerable slow-down network, they will reduce or even desist in their use, being restraint users, and freeing up bandwidth that can once again be consumed, and so on. In this way, every individual user can adjust its use to adapt network circumstance perceived by it. The resources required by users or the measurements of macroscopic performance of the network are regarded as network entities in Fig.4. The entry \bar{a}_{ij} of the M in Fig.4 represents the i -th class of resources occupied by j -th user, or i -th performance perceived by j -th user. The network dynamics is involved in the network state perceived by users and the users' attitude for using networks.

Example 4. The network dynamics associated with TCP adaptive congestion window.

Instead of the users of Example 3, the TCP sources exhibit an adaptive behavior during their segment generation processes through dynamically changing their congestion window sizes to adapt to the current network status that they perceive and estimate. As mentioned in Example 3, we can use the GCA to model the network dynamics caused by such TCP sources' adaptive behaviors.

Example 5. The network dynamics given rise to by parallel optimization. We consider such problem-solving in a multi-agent systems (MAS) built on the network that each agent tries to increase its own personal utility via cooperation with the whole outcome of the system increased as much as possible and with the some constraints satisfied as better as possible. Like the Example 2, the agents and tasks are also regarded as the entities and requests of the M in Fig.4, respectively. Each \bar{a}_{ij} in Fig.4, as a primitive macro-cell, has its own autonomous dynamic equation with performance feedback from the MAS. Generally, each row $A_i = (\bar{a}_{i1}, \dots, \bar{a}_{im})$ or column $T_j = (\bar{a}_{1j}, \dots, \bar{a}_{nj})$ forms a macro-cell with larger granularity than macro-cell \bar{a}_{ij} . As a large number of macro-cell dynamics progress parallelly, the MAS in networks will yield the corresponding aggregate dynamic behavior.

We conclude as follows. The proposed GCA approach can model the dynamic behavior of computer networks, which fully consider the autonomy of network entities, the social interactions among individuals, the macroscopic performance of different granularity colonies, and the influences of concurrent emergent events in the networks. The simulation on the network dynamic behaviors shows the effectiveness and validation of the proposed GCA architecture and algorithm.

References

1. Gupta, A., Stahl, D., Whinston, A.: Priority Pricing of Integrated Services Networks. Internet Economics. MIT Press, Cambridge (1997)
2. Shenker, S., Clark, D., Estrin, D., Herzog, S.: Pricing in Computer Networks: Reshaping the Research Agenda. Computer Communications Review 26(2), 19–43 (1996)
3. Forti, M., Tesi, A.: New Conditions for Global Stability of Neural Networks with Application to Linear and Quadratic Programming Problems. IEEE Trans. on Circuits and Systems 42(7), 354–366 (1995)
4. Chua, L.O., Roska, T.: The CNN Paradigm. IEEE Trans. on Circuits and Systems 40(3), 47–156 (1993)
5. Chua, L.O., Yang, L.: Cellular Neural Networks: Theory. IEEE Trans. on Circuits and Systems 35(10), 1257–1272 (1988)
6. Takahashi, N., Chua, L.O.: A New Sufficient Condition for Nonsymmetric CNN's to Have a Stable Equilibrium Point. IEEE Trans. on Circuits and Systems 44(11), 1092–1095 (1997)
7. Arik, S., Tavsanoglu, V.: On the Global Asymptotic Stability of Delayed Cellular Neural Networks. IEEE Trans. on Circuits and Systems 47(4), 571–574 (2000)
8. Grassi, G.: A New Approach to Design Cellular Neural Networks for Associative Memories. IEEE Trans. on Circuits and Systems 44(9), 835–838 (1997)

9. Brucoli, M., Carnimeo, L., Grassi, G.: Discrete-Time Cellular Neural Networks for Associative Memories with Learning and Forgetting Capabilities. *IEEE Trans. on Circuits and Systems* 42(7), 396–399 (1995)
10. Liu, D., Michel, A.N.: Cellular Neural Networks for Associative Memories. *IEEE Trans. on Circuits and Systems* 40(2), 119–121 (1993)
11. Apostolou, N., King, R.E.: A Direct Learning Law for a Class of Auto-Associative Dynamic Neural Networks. *IEEE Trans. on Circuits and Systems* 45(5), 580–583 (1998)
12. Mladenov, V.M., Leenaerts, D.M., Uhlmann, F.H.: Estimation of the Basin of Attractions in CNN's. *IEEE Trans. on Circuits and Systems* 45(5), 571–574 (1998)
13. Casetti, C., Meo, M.: A New Approach to Model the Stationary Behavior of TCP Connections. *IEEE Infocom.*, 367–375 (2000)
14. Durbin, R., Willshaw, D.: An Analogue Approach to the Travelling Salesman Problem Using an Elastic Net Method. *Nature* 326, 689–691 (1997)
15. Wong, W.S., Funkalea, C.A.: An Elastic Net Solution to Obstacle Avoidance Tour Planning. In: *Proc. IEEE Int. Conf. on Neural Networks*, vol. III, pp. 799–804. IEEE Press, Los Alamitos (1993)
16. Burr, D.J.: An Improved Elastic Net Method for the Traveling Salesman Problem. In: *Proc. IEEE Int. Conf. on Neural Networks*, vol. I, pp. 69–76. IEEE Press, San Diego (1988)

Optimal Control of Switched System Based on Neural Network Optimization

Rong Long¹, Jinming Fu^{1,*}, and Liyan Zhang²

¹ School of Science, Huazhong Agricultural University, Wuhan, China, 430070
llhau@sohu.com

² School of Automation, Wuhan University of Technology, Wuhan, China, 430070

Abstract. The optimal control problem of switched system is to find both the optimal control input and optimal switching signal and is a mixed integer problem. High computational burden in solving this problem is a major obstacle. To solve this problem, this paper presented hybrid neural network combining continuous neurons and discrete neurons and designed Lyapunov function to guarantee the convergency of proposed hybrid neural network. This new solution method is more suitable to parallel implementation than the mathematical programming. Simulation results show that this approach can utilize fast convergence property and the parallel computation ability of neural network and apply to real-time control.

1 Introduction

The optimal control problem of switched control system is one of the challenging research topics and has been attracting much attention in the control community. Switched linear systems belong to a special class of hybrid control systems, which comprises a collection of subsystems described by linear dynamics (differential/difference equations), together with a switching rule that specifies the switching between the subsystems [1]. Such systems can be used to describe a wide range of physical and engineering systems in practice. Switched linear systems not only provide a challenging forum for academic research, but also bridge the gap between the treatment of linear systems and that of highly complex and/or uncertain systems.

For the switched linear control system, the aim of optimal control is to seek appropriate switching/control strategies to optimize a certain performance index. If the switching signal is given and fixed, the switched system is in fact a time-varying control system. Hence, the problem is reduced to the conventional optimal control problem and can be addressed using either the classic maximum principle or the dynamic programming approach. However, if the switching signal is a design variable or generated by an event-driven (state feedback) switching device, complication arises. We can find an optimal solution by doing backward recursion of the cost using dynamic programming, evaluating all possible choices of switching signals. However, the searching will grow exponentially with optimal time. To avoid combinatoric explosion, a method proposed for efficient pruning of the search tree in [2]. Though

* Corresponding author.

the algorithm greatly reduces the searching time, it still runs in an exponential time. In [3] a method is used to transcribe an optimal control problem into an equivalent problem and then obtains the values of the derivatives based on the solution of a two point boundary value differential algebraic equation formed. An embedding technique can be used to reduce the hybrid optimal control problem to the traditional one and does not require mixed integer programming methods in [4]. In [12] two different approaches are presented for solving optimal control problem of switched system. The first approach iterates between a procedure that finds an optimal switching sequence of modes, and a procedure that finds the optimal switching instants. The second approach is inspired by dynamic programming and identifies the regions of the state space.

In solving optimal problems, one has the alternative of using an electrical circuit which simulates both the objective and constraint functions [5]. This approach was first proposed by Dennis in 1959 [6] and the basic building blocks used in the quadratic programming circuit are dc voltage and current sources, ideal diodes and multiport transformers. Tank and Hopfield proposed the first working recurrent neural network (RNN) implemented on analog circuits [7, 8]. At present, there are several RNN approaches to solving optimal problems. Moreover, techniques based on the Lagrangian function obtain near optimal solutions to mixed integer problems with equality constraints. In [9] a coupled gradient network approach is proposed for a class of constraint mixed-integer optimization. In [10], Lagrangian augmented Hopfield network is constructed by including augmented Lagrangian multiplier neurons in the augmented Hopfield network.

To solve optimal control of switched system, this paper present hybrid neural network combining continuous neurons and discrete neurons. The organization of this paper is as follows. Section 2 presents the model of switched system and the structure of optimal control. In section 3 we advance the hybrid neural network to solve the mixed integer problem. In section 4 a numerical example is simulated to show the good performance of the proposed methods and Section 5 concludes this paper.

2 Optimal Control of Switched System

It is assumed that a piecewise-affine system and a corresponding partition of the state space with polytopic cells R_i , $i \in I = \{1, 2, \dots, m\}$ are given. Moreover, the set R_i partition R such that $\bigcup_{i=1}^m \bar{R}_i = R$, $R_i \cap R_j = \emptyset$, $i \neq j$, where \bar{R}_i denotes the closure of R_i .

Mathematically, a discrete switched linear control system can be described by

$$x(k+1) = A_i x(k) + B_i u(k) \quad (1-1)$$

$$y(k) = C_i x(k) \quad (1-2)$$

where $x \in R^n$ is the state variable, $u \in R^p$ is the control input, $y \in R^q$ is the output, i is the piecewise constant switching signal taking value from the finite index set $I = \{1, 2, \dots, m\}$.

Given the structure of each individual subsystem, the overall system behavior is determined by the switching rule of switching signal. In general, the switching rule of switching signal may depend on its past value of the state/output.

$$i(k+1) = \Psi(x(k), y(k), u(k)) \tag{2}$$

Consider the constrained finite-time optimal control problem

$$\min_u J = \sum_{j=1}^N (y(k+j) - y_r)^T Q (y(k+j) - y_r) + \sum_{j=0}^{N-1} (u(k+j) - u_r)^T P (u(k+j) - u_r) \tag{3-1}$$

$$\text{s.t. } x(k+1) = A_i x(k) + B_i u(k) \tag{3-2}$$

$$y(k) = C_i x(k) \tag{3-3}$$

$$i(k+1) = \Psi(x(k), y(k), u(k)) \tag{3-4}$$

$$u_{\min} \leq u(k) \leq u_{\max} \quad j = 0, \dots, p-1 \tag{3-5}$$

where N is recede horizon, y_r and u_r are reference trajectory of output variable and manipulated variable, respectively. $Q \in R^{q \times q}$ and $P \in R^{p \times p}$ are weight matrices. $\Psi(x(k), y(k), u(k))$ is switching rule of switching signal.

In this optimal control problem there are m model and switching rules. It is hard to solve this problem. We define binary variable $v(k) = [v_1, v_2, \dots, v_m] \in \{0, 1\}$ and rewrite (3-2) and (3-3) as

$$x(k+1) = \sum_{i=1}^m A_i x(k) v_i(k) + \sum_{i=1}^m B_i u(k) v_i(k) \tag{4-1}$$

$$y(k) = \sum_{i=1}^m C_i x(k) \tag{4-2}$$

$$\sum_{i=1}^m v_i(k) - 1 = 0 \tag{4-3}$$

Substitute these equations (4-1)-(4-4) into optimal control problem (3), we can get

$$\min_{U, V} J = f(U, V) = \frac{1}{2} U(k)^T \Gamma U(k) + E^T U(k) + M V(k) \tag{5-1}$$

$$\text{s.t. } i(k+1) = \Psi(x(k), y(k), u(k)) \tag{5-2}$$

$$\sum_{i=1}^m v_i(k) - 1 = 0 \tag{5-3}$$

$$U_{\min} \leq U(k) \leq U_{\max} \tag{5-4}$$

where

$$U(k) = [u(k) \ u(k+1) \ \dots \ u(k+N-1)]^T$$

$$V(k) = [v(k) \ v(k+1) \ \dots \ v(k+N-1)]^T$$

Γ , E and M are coefficient matrix.

So each sample instant of optimal control problem (5) is a mixed integer problem. To solve this problem we present hybrid neural network approach in next section.

3 Hybrid Neural Network

In this section a hybrid neural network is constructed by combining continuous neurons with discrete neurons. This hybrid neural network consists of two interacting networks: a network to represent the real-valued variables, called the *r*-network, and a network to represent the binary valued variables, called the *b*-network. Hence the state space for this dynamical system will be hypercube given by

$$\Omega = \prod_{i=1}^L [m_i, M_i] \times \prod_{i=1}^M [0,1]$$

where $L = p \times (N - 1)$, $M = m \times (N - 1)$.

The input to the *i*th node in the *r*-network will be denoted by h_i^u , and input to the *i*th node in the *b*-network will be denoted by h_i^v . Later the dynamics of the coupled net will be defined in terms of these input variables.

To ensure that $u_i \in [m_i, M_i]$ and $v_i \in \{0,1\}$, the activation function is defined as follow

$$u_i = P_i(h_i^u) = \begin{cases} m_i & h_i^u < m_i \\ h_i^u & m_i \leq h_i^u \leq M_i \\ M_i & h_i^u > M_i \end{cases}, \quad i = 1, 2, \dots, L \tag{6}$$

$$v_i = \Phi_i(h_i^v) = \begin{cases} 0 & h_i^v < 0 \\ h_i^v & 0 \leq h_i^v \leq 1 \\ 1 & h_i^v > 1 \end{cases}, \quad i = 1, 2, \dots, M \tag{7}$$

The energy function for this network will be constructed using a penalty function approach. A penalty function enforces the constraint and switching rule and comprised of three terms:

$$P(u, v) = G(u, v) + H(u, v) + K(v) \tag{8}$$

In penalty function the first penalty term is responsible for enforcing the equality constraints, $G(u, v)$ will take the following form:

$$G(u, v) = \left(\sum_{i=1}^M v_i - 1 \right)^2 \tag{9}$$

The second term adds a positive penalty if the solution (u, v) does not satisfy the switching rule, $H(u, v)$ will take the following form:

$$H(u, v) = \begin{cases} 1 & \text{if solution does not satisfy (5-2)} \\ 0 & \text{if solution satisfy (5-2)} \end{cases} \tag{10}$$

Finally, we require $v_i \in \{0,1\}$ for all $i = 1, 2, \dots, m$. This constraint will be captured by the following penalty function.

$$K(v) = \sum_{i=1}^M v_i(1 - v_i) \tag{11}$$

Suppose $(u, v) \in \Omega$ is a feasible point of mixed integer problem. All the constraints are satisfied, then clearly, $G(u, v) = H(u, v) = V(v) = 0$.

The activation function used for this network will guarantee that the constraint $(u, v) \in \Omega$ is satisfied. Finally, the energy function for the network is given as follow

$$E(u, v) = Af(u, v) + B\left(\sum_{i=1}^M v_i - 1\right)^2 + CH(u, v) + D\sum_{k=1}^M v_k(1 - v_k) \tag{12}$$

where A, B, C and D are positive penalty coefficients.

The dynamics for the network are obtained by gradient descent on the energy function. The dynamic equations are given as follows

$$\dot{h}_i^u = -\eta_u \frac{\partial E}{\partial u_i} = -\eta_u A \frac{\partial f}{\partial u_i} \tag{13}$$

$$u_i = P_i(h_i^u) \quad i = 1, \dots, L \tag{14}$$

$$\dot{h}_i^v = -\eta_v \frac{\partial E}{\partial u_i} \tag{15}$$

$$= -\eta_v \left[A \frac{\partial f}{\partial v_i} + 2B \left(\sum_{j=1}^m v_j - 1 \right) + CH + D(1 - 2v_i) \right]$$

$$u_i = \Phi_i(h_i^v) \quad i = 1, \dots, M \tag{16}$$

where η_u and η_v are positive coefficients which will be used to scale the dynamics of the hybrid neural network.

It is now shown that the dynamics of the system cause the energy function to decrease. First consider the rate of change of the energy function due to a change in the output of continuous neurons and discrete neurons.

$$\begin{aligned} \frac{dE}{dt} &= \sum_{i=1}^L \frac{\partial E}{\partial u_i} \dot{u}_i + \sum_{i=1}^M \frac{\partial E}{\partial v_i} \dot{v}_i \\ &= -\sum_{i=1}^L \eta_u (h_i^u)^2 P_i'(h_i^u) - \sum_{i=1}^M \eta_v (h_i^v)^2 \Phi_i'(h_i^v) \leq 0 \end{aligned} \tag{17}$$

Hence energy function E is a Lyapunov function for the network and monotonically decreases until a local minimum of E is reached. This new solution method is more suitable to parallel implementation than the mathematical programming.

4 Numerical Example

In this section, simulation results demonstrate the effectiveness and efficiency of the proposed optimal control of switched system based on hybrid neural network. The simulation is conducted in MATLAB environment.

Consider the 2-dimensional piece-wise linear system [11].

$$x(k+1) = A_i x(k) + B_i u(k)$$

$$y(k) = C_i x(k)$$

Such that:

$$i = \begin{cases} 1, & \text{if } x_1(k) \geq 0 \ \& \ x_2(k) \geq 0 \\ 2, & \text{if } x_1(k) \leq 0 \ \& \ x_2(k) \geq 0 \\ 3, & \text{if } x_1(k) \leq 0 \ \& \ x_2(k) \leq 0 \\ 4, & \text{if } x_1(k) \geq 0 \ \& \ x_2(k) \leq 0 \end{cases}$$

$$A_1 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, B_1 = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}, A_2 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, B_2 = \begin{bmatrix} -1 \\ -0.5 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}, B_3 = \begin{bmatrix} -1 \\ 0.5 \end{bmatrix}, A_4 = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}, B_4 = \begin{bmatrix} 1 \\ -0.5 \end{bmatrix}$$

$$C_1 = C_2 = C_3 = C_4 = [1 \ 0],$$

Reference trajectory of output variable is $y_r = 1$. The state and input constraints, respectively, are: $-5 \leq x_1(k) \leq 5, -5 \leq x_2(k) \leq 5, -1 \leq u(k) \leq 1$.

At first we simulate the convergence of proposed hybrid neural network at single sample instant. The penalty coefficients A, B, C and D were determined empirically by running trial simulations: $A=1, B=5000, C=1000, D=10$. The scaling factors were set at $\eta_u = 3, \eta_v = 0.4$, initial point of control input is $u_0 = 0.1$ and the start point of state variable is $x_0 = [0, 0]^T$. The simulation result of control input u is shown in Fig. 3. From Fig. 3 we see that convergence time is both 0.5. So this hybrid neural network can complete the computation during the sample instant.

Then we simulate the optimal control of this switched system by using hybrid neural network optimization. The penalty coefficients A, B, C and D are $A=1, B=5000,$

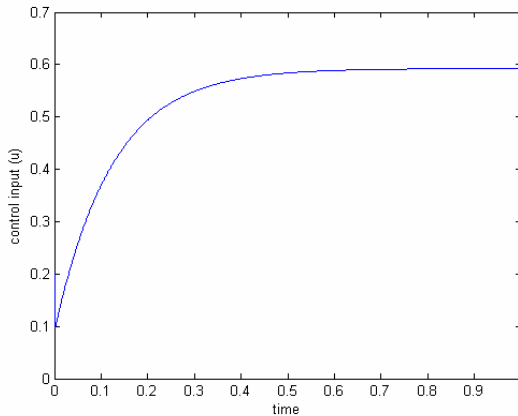


Fig. 3. trajectory of control input at single instant

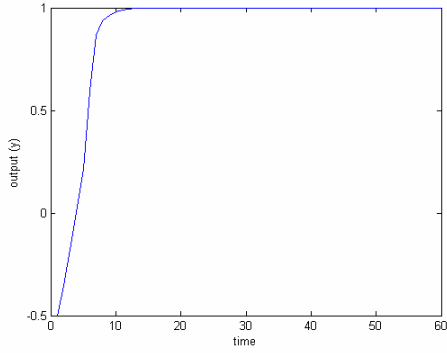


Fig. 4. simulation result of output y

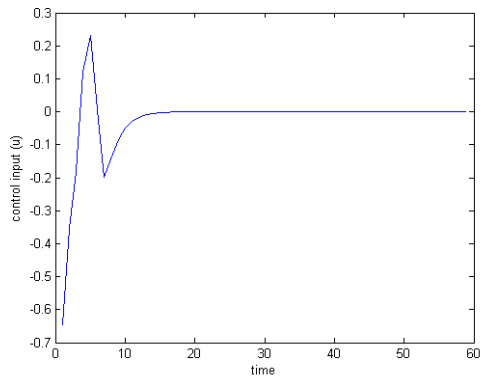


Fig. 5. Simulation result of control input u

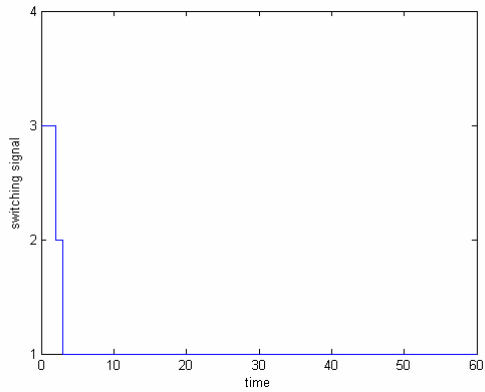


Fig. 6. Simulation result of switching signal

$C=1000$, $D=10$. The scaling factors were set at $\eta_u = 3$, $\eta_v = 0.4$, initial point of control input is $u_0 = 0.1$ and the start point of state variable is $x_0 = [-0.5, 0.5]^T$. The simulation results of output y , control input u and switching signal are shown in Fig. 4-6. From the simulation result demonstrate the effectiveness and efficient of proposed optimal control of switched system based on hybrid neural network and can to real-time control.

5 Conclusion

In this paper we present hybrid neural network combining continuous neurons and discrete neurons. We define energy function of this network and designed lyapunov function to guarantee the convergency of proposed hybrid neural network. This new solution method is more suitable to parallel implementation than the mathematical programming. Simulation results show that this hybrid neural network can complete the computation during the sample instant and this approach can utilize fast converge property and the parallel computation ability of neural network and apply to real-time control.

References

1. Sun, Z.D., Ge, S.S.: Analysis and Synthesis of Switched Linear Control Systems. *Automata* (41), 181–195 (2005)
2. Lincoln, B., Bernhardsson, B.: LQR Optimization of Linear System Switching. *IEEE Transactions on Automatic Control* 47(10), 1701–1705 (2002)
3. Xu, X.P., Antsaklis, P.J.: Optimal Control of Switched Systems Based on Parameterization of the Switching Instants. *IEEE Transactions on Automatic Control* 49(1), 2–16 (2004)
4. Wei, S.M., Uthaichana, K., et al.: Applications of Numerical Optimal Control to Nonlinear Hybrid Systems. *Nonlinear Analysis: Hybrid Systems* 1(2), 264–279 (2007)
5. Kennedy, M.P., Chua, L.O.: Neural Networks for Nonlinear Programming. *IEEE Transaction on Circuits and Systems* 35(5), 554–562 (1988)
6. Xia, Y., Wang, J.: A Dual Neural Network for Kinematic Control of Redundant Robot Manipulators. *IEEE Transactions on Systems, Man and Cybernetics, Part B* 31(1), 147–154 (2001)
7. Tank, D.W., Hopfield, J.J.: Simple Neural Optimization Networks: An A/D Converter, Signal Decision Circuit, and a Linear Programming Circuit. *IEEE Trans. Circuits Syst.* 33, 5533–5541 (1986)
8. Hopfield, J.J., Tank, D.W.: Computing with Neural Circuits: A Model. *Science* 233, 625–633 (1986)
9. Watta, P.B., Hassoun, M.H.: A Coupled Gradient Network Network Approach for Static and Temporal Mixed-Integer Optimization. *IEEE Transactions on Neural Networks* 7(3), 578–593 (1996)
10. Joseph, D.D., Mark, J.O.: A Lagrangian Augmented Hopfield Network for Mixed Integer non-linear Programming Problems. *Neurocomputing* 42, 323–330 (2002)
11. Grieder, P., Kvasnica, M., Baotic, M., Morari, M.: Low Complexity Control of Piecewise Affine Systems with Stability Guarantee. In: *Proceeding of the American Control Conference*, pp. 1196–1201 (2004)
12. Seatzu, C., Corona, D., Giua, A., Bemporad, A.: Optimal Control of Continuous-Time Switched Affine Systems. *IEEE Transactions on Automatic Control* 1(5), 726–741 (2006)

Zhang Neural Network Versus Gradient Neural Network for Online Time-Varying Quadratic Function Minimization

Yunong Zhang¹, Zhan Li¹, Chenfu Yi¹, and Ke Chen²

¹ School of Information Science and Technology

² School of Software

Sun Yat-Sen University, Guangzhou 510275, China

ynzhang@ieee.org, zhynong@mail.sysu.edu.cn

Abstract. With the proved efficacy on solving linear time-varying matrix or vector equations, Zhang neural network (ZNN) could be generalized and developed for the online minimization of time-varying quadratic functions. The minimum of a time-varying quadratic function can be reached exactly and rapidly by using Zhang neural network, as compared with conventional gradient-based neural networks (GNN). Computer-simulation results substantiate further that ZNN models are superior to GNN models in the context of online time-varying quadratic function minimization.

1 Introduction

The problem of solving quadratic-minimization is widely encountered in science and engineering areas [1][2]. It could be an important part of many solution procedures; e.g., signal-processing [1] and robot motion planning [2][3]. There are two general types of solution to the problem of quadratic minimization. One is the numerical algorithms [2][4] performed on digital computers (i.e., on our today's computers). Usually, the minimal arithmetic operations of a numerical algorithm are proportional to the cube of the Hessian matrix's dimension, and consequently such a numerical algorithm may not be efficient enough for large-scale online applications. In view of this, some $O(n^2)$ -operation algorithms were proposed to remedy this computational problem, e.g., in [1][5][6]. However, they may still not be fast enough; e.g., in [6], it takes on average about one hour to invert a 60000-dimensional matrix. Being the second general type of solution, many parallel-processing computational methods have been proposed, developed, analyzed, and implemented on specific architectures, e.g., the neural-dynamic and analog solvers [3][7]. Such a neural-dynamic approach is now regarded as a powerful alternative for real-time computation in view of its parallel-processing distributed nature and convenience of hardware implementation [3][7][8].

Different from the gradient-based neural networks (in short, gradient neural networks) and most computational algorithms designed intrinsically for time-invariant problems solving [1][3][4][7][9]-[11], a special kind of recurrent neural

networks has recently been proposed by Zhang *et al* for real-time solution of time-varying matrix and vector equations [12] [13]. In this paper, Zhang *et al*'s design method is generalized to solving online the quadratic-minimization problem with time-varying coefficients. Theoretical and simulation results both demonstrate the efficacy of the resultant Zhang-neural-network model. To the best of our knowledge, there is little work dealing with such a time-varying quadratic minimization problem in the neural-network literature at present stage. The main contributions of the paper lie in the following facts.

- 1) In our paper, we propose a special kind of recurrent neural network to solve the time-varying quadratic-minimization problem in real time t .
- 2) Our paper investigates not only the ZNN-model description, but also its theoretical derivation and analysis. In other words, our approach could be used as a systematic approach to solving a set of time-varying problems.
- 3) To show the novelty and difference, we compare the ZNN model with the conventional gradient neural network by solving the same time-varying quadratic minimization problem. The comparison substantiates the efficacy of the ZNN model and shows the less favorable property of the GNN model when used in the time-varying quadratic-minimization.

2 Problem Formulation and Neural Solvers

Consider the following the time-varying quadratic minimization problem:

$$\text{minimize } f(x) := x^T(t)P(t)x(t)/2 + q^T(t)x(t) \in \mathbb{R}, \tag{1}$$

where given Hessian matrix $P(t) \in \mathbb{R}^{n \times n}$ is smoothly time-varying and positive-definite for any time instant $t \in [0, +\infty) \subset \mathbb{R}$, and given coefficient vector $q(t) \in \mathbb{R}^n$ is smoothly time-varying as well. In the expression (1), unknown vector $x(t) \in \mathbb{R}^n$ is to be solved all over the time so as to make the value of $f(x)$ always smallest.

As we may recognize or know, solving the time-varying quadratic minimization problem (1) could be done by zeroing the partial-derivative $\nabla f(x)$ of $f(x)$ [1] at every time instant t ; in mathematics,

$$\nabla f(x) := \frac{\partial f(x)}{\partial x} = \mathbf{0} \in \mathbb{R}^n, \quad \forall t \in [0, +\infty). \tag{2}$$

More specifically, it follows from the above that the theoretical time-varying solution $x^*(t) \in \mathbb{R}^n$ to (1), being the minimum point of $f(x)$ at any time instant t , satisfies $P(t)x^*(t) + q(t) = 0$ or here equivalently $x^*(t) = -P^{-1}(t)q(t)$. The theoretical minimum value $f^* := f(x^*)$ of time-varying quadratic function $f(x)$ is thus achieved as $f^* = x^{*T}(t)P(t)x^*(t)/2 + q^T(t)x^*(t)$.

In order to demonstrate the significance of the interesting problem and the visual effects, we could take the following time-varying coefficients $P(t) \in \mathbb{R}^{2 \times 2}$ and $q(t) \in \mathbb{R}^2$ as an example:

$$P(t) = \begin{bmatrix} 0.5 \sin(t) + 2 & \cos(t) \\ \cos(t) & 0.5 \cos(t) + 2 \end{bmatrix}, \quad q(t) = \begin{bmatrix} \sin(t) \\ \cos(t) \end{bmatrix}. \tag{3}$$

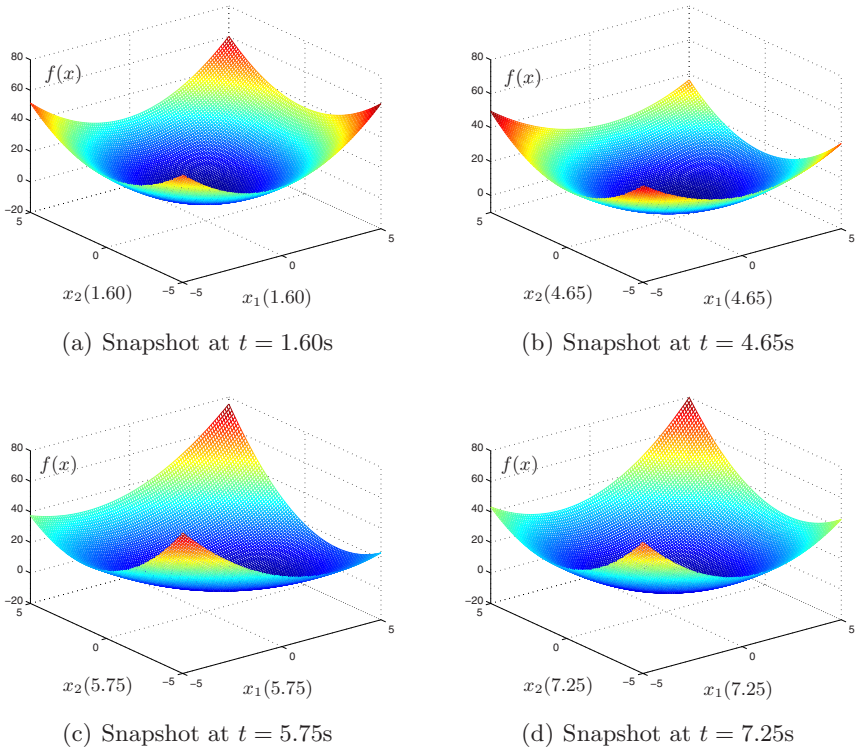


Fig. 1. “Moving” minimum of varying quadratic function $f(x)$ at different time instants

Fig. 1 shows the three-dimensional plots of $f(x)$ with respect to $x = (x_1, x_2) \in \mathbb{R}^{2 \times 2}$, but at different time instants (i.e., $t = 1.60, 4.65, 5.75$ and 7.25s). We can see quite evidently that the shape and minimum value f^* of $f(x)$ together with its minimum solution $x^*(t)$ are all “moving” with time t , so that this time-varying quadratic minimization problem (1) could be considered as a “moving minimum” problem.

2.1 Zhang Neural Networks

In this subsection, Zhang *et al*'s neural-dynamic design method [12] [13] is generalized and applied to solving the online time-varying quadratic minimization problem (1). The design procedure could be formalized as follows.

Step 1: To track the “moving” minimum-point of $f(x)$, instead of transforming the problem to a scalar-valued squared (or norm-based) function to minimize, we could define the vector-valued error function $e(t) \in \mathbb{R}^n$ as ∇f ; in mathematics,

$$e(t) := \nabla f = P(t)x(t) + q(t). \quad (4)$$

Step 2: To make every element $e_i(t)$ of error function $e(t) \in \mathbb{R}^n$ converge to zero (i.e., for any $i \in 1, 2, \dots, n$), the time derivative $\dot{e}(t)$ of error-function $e(t)$ could be constructed better as (or termed, the general ZNN-design formula):

$$\dot{e}(t) := \frac{de(t)}{dt} = \frac{d(\nabla f)}{dt} = -\Gamma\Phi(e(t)) = -\Gamma\Phi(\nabla f), \quad (5)$$

where design parameter $\Gamma \in \mathbb{R}^{n \times n}$ is generally a positive-definite matrix but could be replaced here with any $\gamma > 0 \in \mathbb{R}$ directly to scale the ZNN-convergence rate, and $\Phi(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ denotes an activation-function processing-array from \mathbb{R}^n to \mathbb{R}^n but preferably with each element decoupled. Note that the values of Γ and γ , being a set of reciprocals of capacitance-parameters, should be specified as large as the hardware permits [8]. Constituting the activation-function processing-array $\Phi(\cdot)$, each scalar-valued processing-unit $\phi(\cdot)$ should be odd and monotonically increasing. The following two types of activation function $\phi(\cdot)$ are investigated in this paper: 1) linear activation function $\phi(e_i) = e_i$, and 2) power-sigmoid activation function

$$\phi(e_i) = \begin{cases} e_i^p, & \text{if } |e_i| \geq 1 \\ \frac{1+\exp(-\xi)}{1-\exp(-\xi)} \cdot \frac{1-\exp(-\xi e_i)}{1+\exp(-\xi e_i)}, & \text{otherwise} \end{cases} \quad (6)$$

with suitable design parameters $\xi \geq 2$ and $p \geq 3$.

Step 3: By expanding the ZNN-design formula (5), the following implicit dynamic equation as of Zhang neural network could readily be constructed for minimizing online the time-varying quadratic function (1):

$$P(t)\dot{x}(t) = -\dot{P}(t)x(t) - \gamma\Phi\left(P(t)x(t) + q(t)\right) - \dot{q}(t), \quad (7)$$

where $x(t) \in \mathbb{R}^n$, starting with any initial condition $x(0) \in \mathbb{R}^n$, denotes the neural-state vector which corresponds to the theoretical time-varying minimum solution $x^*(t)$ of non-stationary quadratic function (1). In addition, from (7), it is worth writing out the following linear ZNN model for the same time-varying quadratic function minimization purposes:

$$P(t)\dot{x}(t) = -\left(\dot{P}(t) + \gamma P(t)\right)x(t) - \left(\gamma q(t) + \dot{q}(t)\right). \quad (8)$$

In summary, the designed ZNN models could solve online the time-varying quadratic function minimization problem depicted in (1)! In other words, the ‘‘moving’’ minimum point could be found by the ZNN models in real time and in an error-free manner. For this efficacy, please refer to the ensuing sections.

2.2 Gradient Neural Networks

For comparison, we develop here the conventional gradient-based neural networks to solve online the quadratic minimization problem depicted in (1). However, please note that almost all numerical algorithms and neural-dynamic computational schemes (specially, gradient-based neural networks) [1]-[4] [6]-[11] [14] were

designed intrinsically for the problems with constant coefficients rather than time-varying ones. According to the gradient-descent design method [14], we could obtain the gradient-based neural networks which minimize the stationary quadratic function $f(x) = x^T Px/2 + q^T x$. The design procedure is as follows.

Step 1: Let us define a scalar-valued norm-based (or squared) energy function $\varepsilon(x) = \|\nabla f\|_2^2/2 = \|Px + q\|_2^2/2$, where $\|\cdot\|_2$ denotes the two-norm of a vector. Evidently, x is the minimum solution if and only if $\varepsilon(x) = 0$ is reached.

Step 2: The following design-formula and dynamics of GNN model could then be adopted to minimize online the stationary quadratic function $x^T Px/2 + q^T x$:

$$\dot{x}(t) = \frac{dx(t)}{dt} = -\gamma \frac{\partial \|\nabla f\|_2^2/2}{\partial x} = -\gamma P^T (Px(t) + q), \tag{9}$$

where design parameter $\gamma > 0$ is defined the same as that in ZNN models for the purpose of scaling the neural-network convergence rate. In addition, the following nonlinear GNN model could be extended from the above by exploiting the aforementioned activation-function processing-array $\Phi(\cdot)$:

$$\dot{x}(t) = -\gamma P^T \Phi(Px(t) + q). \tag{10}$$

In summary, the designed GNN models could theoretically only solve the stationary quadratic-function minimization problem (and might be extended in practice to handle approximately the time-varying quadratic function minimization problem). For this point, please refer to the ensuing sections.

3 Theoretical Results and Comparisons

For Zhang neural network [7] which minimizes time-varying quadratic function [1], we could have the following proposition on its global exponential convergence when using the two aforementioned activation-function arrays [11]-[14].

Proposition 1. *Given smoothly time-varying positive-definite matrix $P(t) \in \mathbb{R}^{n \times n}$ and vector $q(t) \in \mathbb{R}^n$, if an odd activation-function array $\Phi(\cdot)$ is employed, then the neural-state vector $x(t)$ of ZNN [7], starting from any initial state $x(0) \in \mathbb{R}^n$, always converges to the theoretical time-varying minimum solution $x^*(t)$ of non-stationary quadratic function [1]. In addition, Zhang neural network [7] possesses the following properties.*

- 1) *If the linear-activation-function array is used, then global exponential convergence could be achieved for ZNN [7] with error-convergence rate γ .*
- 2) *If the power-sigmoid-activation-function array is used, then superior convergence can be achieved for the whole error range $(-\infty, +\infty)$, as compared to the linear-activation-function-array situation.*

For comparison, the following proposition about gradient neural network [10] could be provided but with exactness only for stationary quadratic minimization (i.e., the quadratic minimization with coefficients being constant) [11]-[14].

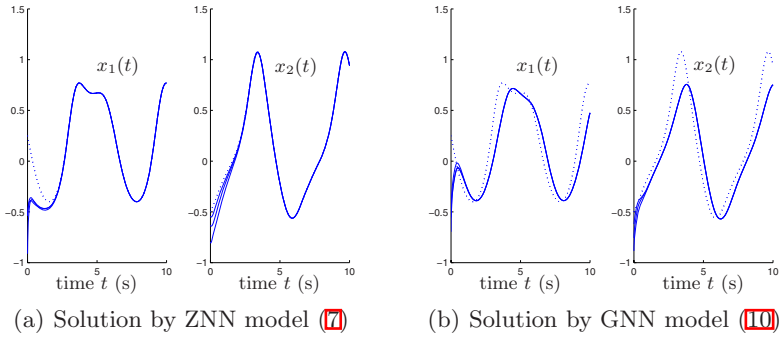


Fig. 2. Time-varying quadratic minimization (1) synthesized online by ZNN (7) and GNN (10) with design parameter $\gamma = 1$ and using power-sigmoid activation functions, where dotted curves correspond to the theoretical time-varying minimum solution $x^*(t)$

Proposition 2. Consider the situation with constant positive-definite matrix $P \in \mathbb{R}^{n \times n}$ and constant vector $q \in \mathbb{R}^n$ associated with quadratic minimization (1). If the linear activation function array is used, then global exponential convergence to the constant minimum point x^* of stationary $f(x)$ can be achieved by GNN (10) with convergence rate proportional to the product of γ and minimum eigenvalue of $P^T P$. If the power-sigmoid activation function array is used, then superior convergence can be achieved for GNN (10) over the whole error range $(-\infty, +\infty)$, as compared to the case of using linear activation functions.

4 Simulation Studies

For comparison, both ZNN model (7) and GNN model (10) are employed to carry out the minimization process and to obtain the time-varying minimum solution $x^*(t)$ of non-stationary quadratic function (1). The power-sigmoid activation function array (with parameters $\xi = 4$ and $p = 3$) is used in the model simulation of both ZNN (7) and GNN (10) with $\gamma = 1$.

- As seen from Fig. 2(a), starting from randomly-generated initial state $x(0)$, neural-state vector $x(t)$ of ZNN (7) could always “elegantly” converge to the time-varying minimum solution $x^*(t)$. On the other hand, as can be seen from Fig. 2(b), relatively large steady-state computational errors of gradient-based neural network (10) exist!
- The ZNN-solution exactness and GNN-solution deviation could be further shown in Fig. 3(a) and (b). We could observe from the left graph of the figure that when proceeding to its steady-state (e.g., after $t \geq 2$), the ZNN model (7) could always compute accurately the minimum value of non-stationary quadratic function $f(x)$ at every time instant t . On the other hand, as seen from the right graph of Fig. 3, though proceeding earlier to its steady-state (e.g., after $t \geq 1.5$), the GNN model (10) generates a result, sometimes equal

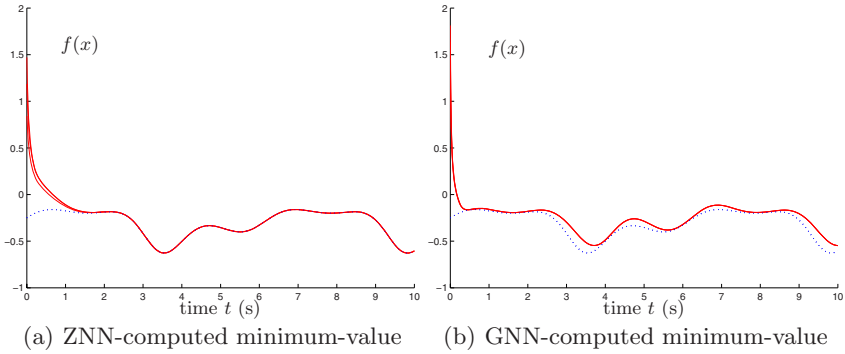


Fig. 3. Time-varying quadratic-function value $f(x)$ minimized online by ZNN (7) and GNN (10) with design parameter $\gamma = 1$ and using power-sigmoid activation functions, where dotted curves correspond to the theoretical time-varying minimum value $f(x^*(t))$

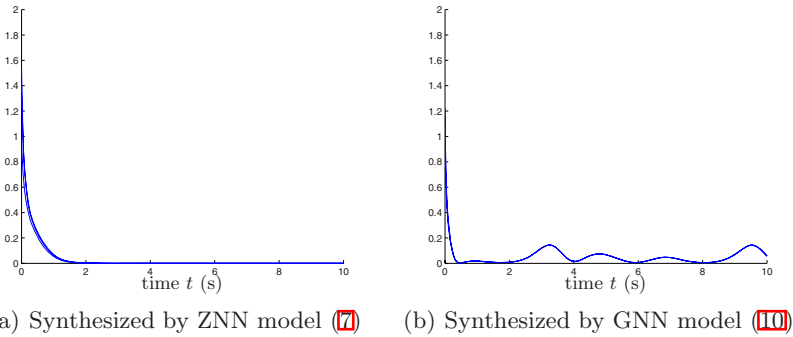


Fig. 4. Residual error $|f(x) - f(x^*)|$ synthesized by ZNN and GNN models with $\gamma = 1$

to or sometimes larger than (i.e., in a hit-and-miss manner) the theoretical minimum value of non-stationary quadratic function $f(x)$ over time t .

- To monitor the convergence properties, we could also show the residual error $|f(x) - f(x^*(t))|$, where $|\cdot|$ denotes the absolute value of a scalar. As seen from Fig. 4(a), $f(x)$ minimized by ZNN (7) could converge perfectly to the theoretical time-varying minimum-value $f(x^*(t))$. In comparison, as seen from Fig. 4(b), $f(x)$ minimized by GNN (10) can not achieve $f(x^*(t))$ exactly.

5 Conclusions

In this paper, a recurrent neural network [i.e., ZNN model (7)] has been established for online time-varying quadratic minimization. such model could generate the moving minimum exactly and efficiently. Computer-simulation results have

substantiated the superior performance and efficacy of ZNN model the investigated non-stationary quadratic minimization problem.

References

1. Leithead, W.E., Zhang, Y.: $O(N^2)$ -Operation Approximation of Covariance Matrix Inverse in Gaussian Process Regression Based on Quasi-Newton BFGS Method. *Communications in Statistics – Simulation and Computation* 36, 367–380 (2007)
2. Tsiotras, P., Corless, M., Rotea, M.: Optimal Control of Rigid Body Angular Velocity with Quadratic Cost. *Journal of Optimization Theory and Applications* 96, 507–532 (1998)
3. Zhang, Y.: Towards Piecewise-Linear Primal Neural Networks for Optimization and Redundant Robotics. In: *Proceedings of IEEE International Conference on Networking, Sensing and Control*, pp. 374–379. IEEE Press, Ft. Lauderdale (2006)
4. Davey, K., Ward, M.J.: A Successive Preconditioned Conjugate Gradient Method for the Minimization of Quadratic and Nonlinear Functions. *Applied Numerical Mathematics* 35, 129–156 (2000)
5. Gomez, M.A.: An $O(n^2)$ Active Set Algorithm for Solving Two Related Box Constrained Parametric Quadratic Programs. *Numerical Algorithms* 27, 367–375 (2001)
6. Zhang, Y., Leithead, W.E., Leith, D.J.: Time-Series Gaussian Process Regression Based on Toeplitz Computation of $O(N^2)$ Operations and $O(N)$ -Level Storage. In: *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 3711–3716. IEEE Press, Sevilla (2005)
7. Zhang, Y., Wang, J., Xia, Y.: A Dual Neural Network for Redundancy Resolution of Kinematically Redundant Manipulators Subject to Joint Limits and Joint Velocity Limits. *IEEE Transactions on Neural Networks* 14, 658–667 (2003)
8. Mead, C.: *Analog VLSI and Neural Systems*. Addison-Wesley, Reading (1989)
9. Manherz, R.K., Jordan, B.W., Hakimi, S.L.: Analog Methods for Computation of the Generalized Inverse. *IEEE Transactions on Automatic Control* 13, 582–585 (1968)
10. Sturges Jr., R.H.: Analog Matrix Inversion (Robot Kinematics). *IEEE Journal of Robotics and Automation* 4, 157–162 (1988)
11. Zhang, Y., Chen, K.: Global Exponential Convergence and Stability of Wang Neural Network for Solving Online Linear Equations. *Electronics Letters* 44, 145–146 (2008)
12. Zhang, Y., Jiang, D., Wang, J.: A Recurrent Neural Network for Solving Sylvester Equation with Time-Varying Coefficients. *IEEE Transactions on Neural Networks* 13, 1053–1063 (2002)
13. Zhang, Y., Ge, S.S.: Design and Analysis of a General Recurrent Neural Network Model for Time-Varying Matrix Inversion. *IEEE Transactions on Neural Networks* 16, 1477–1490 (2005)
14. Zhang, Y.: Revisit the Analog Computer and Gradient-Based Neural System for Matrix Inversion. In: *Proceedings of IEEE International Symposium on Intelligent Control*, pp. 1411–1416. IEEE Press, Limassol (2005)

A Logic Description on Different Negation Relation in Knowledge

Zhenghua Pan

School of Science, Jiangnan University, Wuxi, China
panzh@jiangnan.edu.cn

Abstract. In this paper, we propose that negation relation in knowledge ought to differentiate contradictory negation and opposite negation. Based on this cognition, we (1) discovered a character of knowledge: if pair of knowledge with opposite negation relation are fuzzy knowledge, then must exist fuzzy medium object (new knowledge) between them; contrarily, if there is a fuzzy medium object between the two knowledge with opposite negation relation, then they must be fuzzy knowledge; (2) five kinds of contradictory negation and opposite relations (CDC, CFC, ODC, OFC and ROM) in the distinct knowledge and fuzzy knowledge, and formalization definition were confirmed; (3) studied these different negation relations using the medium predicate logic MF and the infinite valued semantic model Φ of MF, and obtained the conditions of processing these different negation relations.

1 Introduction

In computational information processing systems, classical logic has already been basis for negations and their relations in knowledge at all times. Recently, with development of knowledge processing research, some scholars consider that there need two kinds of negations in the field of knowledge processing. G. Wagner etc proposed that from a logic point of view, negation is not a clean concept in computational information processing systems such as knowledge-based reasoning, natural language, logic programming languages (such as Prolog), semantic web, imperative programming languages (such as Java), database query languages (such as SQL), modeling languages (such as UML/OCL) and in production rule systems (such as CLIPS and Jess), and pointed out there are two kinds of negation in above systems: a strong negation expressing explicit falsity and a weak negation expressing non-truth [1]-[6]. K. Kaneiwa proposed that there were two kinds of negation in description logic, and presented an extended description logic ALC_{\neg} with classical negation and strong negation. In particular, he adhered to the notions of contraries, contradictories and subcontraries, generated from conceivable statement types using predicate denial (e.g., not happy) and predicate term negation (e.g., unhappy). To capture these notions, its formalization provides an improved semantics that suitably interprets various combinations of classical negation and strong negation. And it showed that these semantics preserved contradictoriness and contrariness for ALC_{\neg} -concepts [7]. S. Ferré introduced the epistemic extension, a logic transformation based on the modal logic AIK

(All I Know) for use in the framework of Logical Concept Analysis (LCA), and its aim was to allow for the distinction between negation, opposition, and possibility in a unique formalism. Furthermore, this epistemic extension entailed no loss of generality in LCA [8]. Zhenghua Pan proposed that one new cognition and processing for inconsistency in knowledge, and consider that the study of inconsistent knowledge should differentiate contradictory negation relation and opposite negation relation in knowledge. Moreover, Zhenghua Pan pointed out that the negation in distinct and fuzzy knowledge should contain contradictory negation, opposite negation and fuzzy negation in some opposite knowledge[9],[10],[11].

2 Essence of Negation Concept in Knowledge

Concept is most basic element of knowledge, concept is expressed by language. However, since the attributes of many things have indefiniteness or the descriptive diction of attributes has fuzziness, bring on extensions of many concepts are not distinct and much knowledge becomes fuzzy knowledge. So, we divide concept into the distinct concept (unambiguous intention, distinct extension) and fuzzy concept (unambiguous intention, non-distinct extension). And what also reflects in knowledge science that is *Knowledge should differentiate distinct knowledge and fuzzy knowledge*.

Relations between the two concepts, in fact, had been extensional relations between them and contained consistent relation and inconsistent relation in theory of formal logic. Using a rectangle to denote the extension of a concept, thus the relation between concept A and B is inconsistent if extensions of A and B have not common area.

Inconsistent relation between two concepts has been differentiated into the contradictory relation and opposite relation since Aristotle. In fact, the relation between one concept and its negation is an inconsistent relation, therefore, this relation contains the contradictory negation relation and opposite negation relation. And what also reflects in knowledge science that is *Negation of knowledge contains contradictory negation and opposite negation in knowledge*.

2.1 Five Kinds of Contradictory Negation Relation and Opposite Negation Relation

For relation between one concept and its negation, we present the following five kinds of contradictory relation and opposite relation.

(2.1) *Contradictory Negation Relation in Distinct Concepts CDC:*

Character of CDC: “bound definitude”, “either this or that”.

For example, “rational number” and “irrational number” in concept of genus “real number”; “white” and “non-white” in concept of genus “color”.

(2.2) *Contradictory Negation Relation in Fuzzy Concepts CFC:*

Character of CFC: “bound non-definitude”, “either this or that”.

For example, “quick” and “not quick” in concept of genus “speed”; “young people” and “non-young people” in concept of genus “people”.

(2.3) *Opposite Negation Relation in Distinct Concepts ODC:*

Character of ODC: “bound definitude”, “either this or that” doesn’t holds.

For example, “white” and “black” in concept of genus “color”; “positive number” and “negative number” in concept of genus “number”.

(2.4) *Opposite Negation Relation in Fuzzy Concepts OFC:*

Character of OFC: “bound non-definitude”, “either this or that” doesn’t holds.

For example, “speed” and “slow” in concept of genus “velocity”; “young people” and “old people” in concept of genus “people”.

In multifarious knowledge of world, a lot of opposite concepts have a common character: there is the medium state between the two opposite sides. These medium states have property which is “both this and that”, where “this” and “that” drive at two opposite sides in an opposite concepts. After analysis and research for many instances, we discover that medium concept and opposite concepts have a relation: if there is a medium concept between the two opposite concepts, the opposite concepts and medium concept must be fuzzy concepts, conversely, if the opposite concepts are fuzzy concepts, there must exists a medium concept in between them. That is to say, the two opposite concepts are fuzzy concepts if and only if there is a medium concept between the two opposite concepts. And what also reflects in knowledge science that is *Opposite knowledge is fuzzy knowledge if and only if there is a medium state (new knowledge) between opposite knowledge.*

So, for study of opposite concepts, we must investigate relation between the opposite concepts and the medium concept.

(2.5) *Relation between Opposite concepts and the Medium concept ROM:*

Character of ROM: “bound non-definitude”, all of them are fuzzy concepts.

For example, “young people” and “old people” are opposite concepts, “middle-aged people” is their medium concept; “daylight” and “night” are opposite concepts, “dawn” is their medium concept.

From above, there exist five kinds of contradictory relation and opposite relation between one concept and its negation, and what also reflects in knowledge science that is *There exist five kinds of contradictory relation and opposite relation between distinct and fuzzy knowledge and their negation, that is CDC, ODC, CFC, OFC and ROM.*

2.2 Formalization Definition of Contradictory Negation and Opposite Negation Relations in Knowledge

Since the extension of a concept is that it reflects all of objects and relations between concepts are relations between their extensions, we give, from conceptual extensional point of view, the formalization definition of CDC, CFC, ODC, OFC and ROM as follows.

Definition 1: Let $U (\neq \emptyset)$ be domain of individuals. Any $X (X \subseteq U)$ is called a concept in U , and a variety of concepts on U is called the abstract knowledge about U , called knowledge for short. For arbitrary $X \neq \emptyset$, if there exist a partition $\xi: \{X_1, X_2, \dots, X_n\}$, $X_i \subseteq X$, $X_i \neq \emptyset$, $\bigcup_{i=1}^n X_i = X$, $X_i \cap X_j = \emptyset (i \neq j, i, j = 1, 2, \dots, n)$, we call X a concept of genus on X_1, X_2, \dots, X_n , and call $X_i (i = 1, 2, \dots, n)$ a concept of species on X . If $X_i \cap X_j \neq \emptyset$, then X_i, X_j are fuzzy concepts.

Since a pair of concepts, which have contradictory relation or opposite relation, are a pair of concepts of species on the same concept of genus, furthermore, CDC, CFC, ODC and OFC are the relations between two concepts of species on the only one concept of genus, respectively. By above definitions, they are different binary relations on $X \times X$ for the concept of genus X , that is, they are subsets of $X \times X$, respectively.

Definition 2: Let $A = \bigcup_{i=1}^n A_i$ be a concept of genus, A_i be a concept of species in A .

1) for a $A_i \subseteq A$, there is unique A_j in A ($i \neq j$), A_j and A_i are contradictory concepts, that is,

$$CDC = \{(A_i, A_j) \mid A_i \neq A_j, A_i \cap A_j = \emptyset, A_i \cup A_j = A\} \subset A \times A,$$

if A_i, A_j are fuzzy concepts, then there is

$$CFC = \{(A_i, A_j) \mid A_i \neq A_j, A_i \cap A_j \neq \emptyset, A_i \cup A_j = A\} \subset A \times A;$$

2) for such A_i , there is A_k in A ($k \neq i, k \neq j, i \neq j$), A_i and A_k are opposite concepts, that is,

$$ODC = \{(A_i, A_k) \mid A_i \neq A_k, A_k \neq A_j, A_i \neq A_j, A_i \cap A_k = \emptyset, A_i \cup A_k \subseteq A\} \subset A \times A,$$

if A_i, A_k are fuzzy concepts, then there is

$$OFC = \{(A_i, A_k) \mid A_i \neq A_k, A_k \neq A_j, A_i \neq A_j, A_i \cap A_k \neq \emptyset, A_i \cup A_k \subseteq A\} \subset A \times A.$$

From above, we see that there exists a medium concept between the two opposite concepts if and only if they are fuzzy concepts. So the relation between the opposite concepts and the medium concept should be a subset of $(X \times X) \times X$.

Definition 3: Let $B = \bigcup_{i=1}^n B_i$ be a concepts of genus, B_i be a concept of species in B . If

$B_i, B_j \subseteq B$ ($i \neq j$) are opposite fuzzy concepts, then there is $B_m \subseteq B$ ($m \neq i, m \neq j$), such that

$$ROM = \{((B_i, B_j), B_m) \mid B_i \neq B_j, B_i \cap B_m \neq \emptyset, B_j \cap B_m \neq \emptyset, B_i \cup B_j \cup B_m \subseteq B\} \subset (B \times B) \times B.$$

3 Logical and Semantic Description on CDC,CFC,ODC,OFC and ROM

3.1 One Kind of Logical Description

In the 19th century, G. Frege called the monadic function, whose value is truth value, a concept, and the relation between concepts was the relation between the monadic predicates in logic, actually. Therefore, we should study the logic theory which is adaptable for research of the concepts and their inconsistent relations (contradictory relation, opposite relation) and efficient to describe and process them from syntax and semantic.

The Medium Logic ML is a kind of non-classical formal system established by Chinese scholars, where the medium predicate logic MF is a subsystem of ML [12],[13],[14]. In this paper, we adopt the formal symbols in medium logic ML.

In MF, let P be a predicate (the property or relation of individual), for any object x in the domain of individuals, if either x completely satisfies P or x completely does not satisfy P , i.e. there does not exist such object which partially satisfies P and partially does not satisfy P , then P is called a distinct predicate and denoted as $\text{dis}P$. For predicate P , if there exists the object x which partially has property P and partially does not has property P , P is called a fuzzy predicate and denoted as $\text{fuz}P$. $\neg P$ denotes the opposite side of P , so P and $\neg P$ represent a pair of opposite predicates. $\neg\neg P$ denotes the contradictory side of P , so P and $\neg\neg P$ represent a pair of contradictory predicates. ML unconditionally acknowledged that for some predicate P , there is object x which partially satisfies P and partially satisfies $\neg P$, such x is called a medium object between P and $\neg P$, and that “ x partially satisfies P ” is denoted as $\sim P(x)$. $\sim P$ is called the fuzzy negation of P , so P and $\sim P$ represent a pair of predicates that own the relation of the fuzzy negation.

It is shown that a concept of genus is divided into contradictory concepts and opposite concepts, that is, the monadic predicate is divided into $\{P, \neg P, \neg\neg P, \sim P\}$ in the medium logic (but the monadic predicate is only divided into $\{P, \neg P\}$ in the mathematics and classical logic, that is, representation of contradictory predicate and opposite predicate do not differentiate), where contradictory monadic predicates $P, \neg P$ represent a pair of the contradictory concepts, opposite monadic predicates $P, \neg\neg P$ denote a pair of the opposite concepts, and fuzzy monadic predicates $\sim P$ denote a medium concept between P and $\neg P$. Therefore, five kinds of the contradictory relation and opposite relation of concepts can be expressed by medium predicate logic as follows.

$$\begin{aligned} \text{CDC} &= \{(P, \neg P) \mid \text{dis}P\}, \text{CFC} = \{(P, \neg P) \mid \text{fuz}P\}, \\ \text{ODC} &= \{(P, \neg\neg P) \mid \text{dis}P\}, \text{OFC} = \{(P, \neg\neg P) \mid \text{fuz}P\}, \\ \text{ROM} &= \{(P, \sim P) \mid \text{fuz}P\}. \end{aligned}$$

3.2 One Kind of Semantic Description

Since a concept is a monadic predicate, data description of concept is shown by the truth value of concept in logic theory. Through investigating theory and application of the medium logic, one had established the three-valued semantic model [15],[16], and the infinite-valued semantic model [17],[18]. We consider that the infinite-valued model Φ of medium predicate logic is optimum for the description and processing of contradictory concepts and opposite concepts, as well as contradictory and opposite relations.

Let Φ be the infinite-valued model of MF, $\Phi: \langle D, \mathfrak{R}_\lambda \rangle$, where $\lambda \in (0, 1)$, D is the domain of individuals, \mathfrak{R}_λ is a λ -valuation of MF. The λ -valuation \mathfrak{R}_λ could be defined as follows [18]:

Definition 4: Let A be a formula of MF, $\lambda \in (0, 1)$. The λ -valuation \mathfrak{R}_λ of A consist of following assignments:

- (1) for each symbol of individual constant in A , assigning an object in D ;
- (2) for each symbol of n -place function in A , assigning an mapping from D^n to D ;

(3) for each symbol of the n-place predicate in A, assigning an mapping from D^n to $[0,1]$, and

(3.1) $\mathfrak{R}_\lambda(A)$ takes only a numerical value in $[0, 1]$ at a time when A is an atomic formula;

(3.2) $\mathfrak{R}_\lambda(A) + \mathfrak{R}_\lambda(\neg A) = 1,$

(3.3) $\mathfrak{R}_\lambda(\sim A) =$

$$\frac{2\lambda - 1}{1 - \lambda} (\mathfrak{R}_\lambda(A) - \lambda) + 1 - \lambda, \text{ when } \lambda \in [\frac{1}{2}, 1) \text{ and } \mathfrak{R}_\lambda(A) \in (\lambda, 1]$$

$$\frac{2\lambda - 1}{1 - \lambda} \mathfrak{R}_\lambda(A) + 1 - \lambda, \text{ when } \lambda \in [\frac{1}{2}, 1) \text{ and } \mathfrak{R}_\lambda(A) \in [0, 1 - \lambda]$$

$$\frac{1 - 2\lambda}{\lambda} \mathfrak{R}_\lambda(A) + \lambda, \text{ when } \lambda \in (0, \frac{1}{2}] \text{ and } \mathfrak{R}_\lambda(A) \in [0, \lambda]$$

$$\frac{1 - 2\lambda}{\lambda} (\mathfrak{R}_\lambda(A) + \lambda - 1) + \lambda, \text{ when } \lambda \in (0, \frac{1}{2}] \text{ and } \mathfrak{R}_\lambda(A) \in (1 - \lambda, 1]$$

$$1/2, \text{ when } \mathfrak{R}_\lambda(A) = 1/2$$

(3.4) $\mathfrak{R}_\lambda(A \rightarrow B) = \text{Max} (1 - \mathfrak{R}_\lambda(A), \mathfrak{R}_\lambda(B)),$ where B is a formula of MF;

(3.5) $\mathfrak{R}_\lambda(A \vee B) = \text{Max} (\mathfrak{R}_\lambda(A), \mathfrak{R}_\lambda(B));$

(3.6) $\mathfrak{R}_\lambda(A \wedge B) = \text{Min} (\mathfrak{R}_\lambda(A), \mathfrak{R}_\lambda(B));$

(3.7) $\mathfrak{R}_\lambda(\forall x P(x)) = \text{Min}_{x \in D} \{ \mathfrak{R}_\lambda(P(x)) \}, \mathfrak{R}_\lambda(\exists x P(x)) = \text{Max}_{x \in D} \{ \mathfrak{R}_\lambda(P(x)) \}.$

Proof: By induction on the number of connectives and quantifiers in A.

Definition 5: Let P be a monadic predicate of MF. In the infinite-value interpretation $\Phi: \langle D, \mathfrak{R}_\lambda \rangle, \mathfrak{R}_\lambda(P(x)) = 1$ means P(x) is true, if and only if any object x in D completely satisfies P. $\mathfrak{R}_\lambda(P(x)) = 0$ means P(x) is false, if and only if any object x in D completely does not satisfy P. $\mathfrak{R}_\lambda(P(x)) \in (0, 1)$ means P(x) is partially true, if and only if there is object x in D partially satisfies P.

Obviously, $\mathfrak{R}_\lambda(P(x))$ denotes the degree, which is that object x in D satisfies the monadic predpicate P. According to Definition 4 and above definition of disP and fuzP, we can obtain the following results:

(I) P is a distinct monadic predicate if $\mathfrak{R}_\lambda(P(x)) = 1$ or $\mathfrak{R}_\lambda(P(x)) = 0.$

(II) P is a fuzzy monadic predicate if $\mathfrak{R}_\lambda(P(x)) \in (0, 1).$ In particular, if $\mathfrak{R}_\lambda(P(x)) \equiv \frac{1}{2}$ (identity), then P is a predicate constant.

For any monadic predicate P, in fact the Definition 4 not only confirmed $\mathfrak{R}_\lambda(P(x)) \in [0, 1]$ and relations among $\mathfrak{R}_\lambda(P(x)), \mathfrak{R}_\lambda(\neg P(x))$ and $\mathfrak{R}_\lambda(\sim P(x)),$ but also confirmed a partition δ by λ for the truth-valued range $[0, 1], \delta$ divided $[0, 1]$ into three disjoint subintervals. By the Definition 4, ranges of $\mathfrak{R}_\lambda(P(x)), \mathfrak{R}_\lambda(\neg P(x))$ and $\mathfrak{R}_\lambda(\sim P(x))$ in partition δ have relations with these subintervals as follows.

<1> When $\lambda \geq \frac{1}{2}, \delta: \{ [0, 1 - \lambda], [1 - \lambda, \lambda], (\lambda, 1] \}.$ Either $\mathfrak{R}_\lambda(P(x)) \in (\lambda, 1]$ or $\mathfrak{R}_\lambda(P(x)) \in [0, 1 - \lambda]$ in $\delta.$ If $\mathfrak{R}_\lambda(P(x)) \in (\lambda, 1]$ then $\mathfrak{R}_\lambda(\neg P(x)) = 1 - \mathfrak{R}_\lambda(P(x)) \in [0, 1 - \lambda]$ and $\mathfrak{R}_\lambda(\sim P(x)) \in [1 - \lambda, \lambda]$ by Definition 4, and $\mathfrak{R}_\lambda(\neg \sim P(x)) = \text{Max} \{ \mathfrak{R}_\lambda(\neg P(x)), \mathfrak{R}_\lambda(\sim P(x)) \} = \mathfrak{R}_\lambda(\sim P(x))$ because $\neg P$ had been defined by $P \rightarrow \sim P$ in ML. In the same way, if

$\mathfrak{R}_\lambda(P(x)) \in [0, 1-\lambda]$ then $\mathfrak{R}_\lambda(\neg P(x)) = 1 - \mathfrak{R}_\lambda(P(x)) \in (\lambda, 1]$, $\mathfrak{R}_\lambda(\neg P(x)) \in [1-\lambda, \lambda]$ and $\mathfrak{R}_\lambda(\neg P(x)) = \text{Max}\{\mathfrak{R}_\lambda(\neg P(x)), \mathfrak{R}_\lambda(\neg P(x))\} = \mathfrak{R}_\lambda(\neg P(x))$.

<2> When $\lambda \leq 1/2$, $\delta: \{[0, \lambda], [\lambda, 1-\lambda], (1-\lambda, 1]\}$. Either $\mathfrak{R}_\lambda(P(x)) \in [0, \lambda)$ or $\mathfrak{R}_\lambda(P(x)) \in (1-\lambda, 1]$ in δ . If $\mathfrak{R}_\lambda(P(x)) \in [0, \lambda)$ then $\mathfrak{R}_\lambda(\neg P(x)) = 1 - \mathfrak{R}_\lambda(P(x)) \in (1-\lambda, 1]$ and $\mathfrak{R}_\lambda(\neg P(x)) \in [\lambda, 1-\lambda]$ by Definition 4, and $\mathfrak{R}_\lambda(\neg P(x)) = \text{Max}\{\mathfrak{R}_\lambda(\neg P(x)), \mathfrak{R}_\lambda(\neg P(x))\} = \mathfrak{R}_\lambda(\neg P(x))$. If $\mathfrak{R}_\lambda(P(x)) \in (1-\lambda, 1]$ then $\mathfrak{R}_\lambda(\neg P(x)) = 1 - \mathfrak{R}_\lambda(P(x)) \in [0, \lambda)$, $\mathfrak{R}_\lambda(\neg P(x)) \in [\lambda, 1-\lambda]$ and $\mathfrak{R}_\lambda(\neg P(x)) = \text{Max}\{\mathfrak{R}_\lambda(\neg P(x)), \mathfrak{R}_\lambda(\neg P(x))\} = \mathfrak{R}_\lambda(\neg P(x))$.

3.3 Processing Condition of CDC, CFC, ODC, OFC and ROM

<1> and <2> show that the endpoint of subintervals of the truth-value range $[0, 1]$ related to λ , and λ is alterable in $(0, 1)$, so the changes of ranges of $\mathfrak{R}_\lambda(P(x))$, $\mathfrak{R}_\lambda(\neg P(x))$ and $\mathfrak{R}_\lambda(\neg P(x))$ are determined by λ . According to above 3.2, the description and processing of truth-value of the contradictory relations and opposite relations CDC, CFC, ODC, OFC and ROM relates to the magnitude and change of λ . Concretely, we have the following results.

a) When $\lambda = 1/2$, the partition δ divides $[0, 1]$ into $\{[0, 1/2], 1/2, (1/2, 1]\}$ by <1> and <2>, and $\mathfrak{R}_\lambda(\neg P(x)) \equiv 1/2$ by Definition 4. Hence, $\mathfrak{R}_\lambda(P(x)) \in [0, 1/2) \cup (1/2, 1]$ and $\mathfrak{R}_\lambda(\neg P(x)) \in [0, 1/2) \cup (1/2, 1]$ for δ . By the (I) and (II), it is shown that both P and $\neg P$ are either distinct monadic predicates or fuzzy monadic predicates. Consequently, there exists following cases $a_1)$ and $a_2)$.

$a_1)$ If $\mathfrak{R}_\lambda(P(x)) = 1$ or $\mathfrak{R}_\lambda(P(x)) = 0$, then P and $\neg P$ are the distinct monadic predicates by (I). This case is able to reflect the opposite relation between the two distinct concepts. That is,

- by the infinite-valued model Φ of the medium predicate logic MF, we can describe and dispose ODC when $\lambda = 1/2$ and $\mathfrak{R}_\lambda(P(x)) = 1$ (or $\mathfrak{R}_\lambda(P(x)) = 0$).

$a_2)$ If $\mathfrak{R}_\lambda(P(x)) \neq 1$ and $\mathfrak{R}_\lambda(P(x)) \neq 0$, then P and $\neg P$ are the fuzzy monadic predicates by (II). Here if $\mathfrak{R}_\lambda(P(x)) \in [0, 1/2)$, that is $\mathfrak{R}_\lambda(\neg P(x)) \in (1/2, 1]$ by Definition 4, then $\mathfrak{R}_\lambda(\neg P(x)) = \text{Max}\{\mathfrak{R}_\lambda(\neg P(x)), \mathfrak{R}_\lambda(\neg P(x))\} = \mathfrak{R}_\lambda(\neg P(x))$ because of $\mathfrak{R}_\lambda(\neg P(x)) \equiv 1/2$. This case is able to reflect contradictory relation between the two fuzzy concepts. That is,

- by the infinite-valued model Φ of the medium predicate logic MF, we can describe and dispose CFC when $\lambda = 1/2$ and $\mathfrak{R}_\lambda(P(x)) \in [0, 1/2)$.

b) In particular, when $\mathfrak{R}_\lambda(P(x)) \equiv 1$ or $\mathfrak{R}_\lambda(P(x)) \equiv 0$ (that is $\mathfrak{R}_\lambda(\neg P(x)) \equiv 0$ or $\mathfrak{R}_\lambda(\neg P(x)) \equiv 1$ by Definition 4), then not only P and $\neg P$ are distinct monadic predicates by (I), but also $\mathfrak{R}_\lambda(P(x))$ and $\mathfrak{R}_\lambda(\neg P(x))$ have character of the two-valued logic, which is “either this or that”. Therefore, this case is able to reflect contradictory relation between the two distinct concepts. That is,

- by the infinite-valued model Φ of the medium predicate logic MF, we can describe and dispose CDC when $\mathfrak{R}_\lambda(P(x)) \equiv 1$ or $\mathfrak{R}_\lambda(P(x)) \equiv 0$.

This shows that the two-valued logic is only one especial state of the medium logic for processing the concept and concept relation.

c) Except a) and b), that is $\lambda \neq 1/2$ and $0 < \mathfrak{R}_\lambda(P(x)) < 1$. By <1>, <2> and Definition 4, we have $\mathfrak{R}_\lambda(P(x))$, $\mathfrak{R}_\lambda(\neg P(x)) \in (0, 1)$, $\mathfrak{R}_\lambda(\neg P(x)) \in [1-\lambda, \lambda]$ or $\mathfrak{R}_\lambda(\neg P(x)) \in [\lambda, 1-\lambda]$, and $\mathfrak{R}_\lambda(\neg P(x)) < \mathfrak{R}_\lambda(P(x)) < \mathfrak{R}_\lambda(\neg P(x))$ or $\mathfrak{R}_\lambda(P(x)) < \mathfrak{R}_\lambda(\neg P(x)) < \mathfrak{R}_\lambda(\neg P(x))$. Thus,

according to (II) and above 2.1, not only P , $\exists P$ and $\sim P$ are all fuzzy monadic predicates, but also this case is able to reflect one opposite relation between the two fuzzy concepts and the relation between the opposite concepts and the medium concept. That is,

- by the infinite-valued model Φ of medium predicate logic MF , we can describe and dispose OFC and ROM when $\lambda \neq \frac{1}{2}$ and $0 < \mathfrak{F}_\lambda(P(x)) < 1$.

References

1. Wagner, G.: A Database Needs Two Kinds of Negation. In: Proc. of MFDBS, pp. 357–371 (1991)
2. Alferes, J.J., Pereira, L.M.: On Logic Program Semantics with Two Kinds of Negation. In: Apt, K. (ed.) Int. Joint Conf. and Symp. on LP, pp. 574–588. MIT Press, Cambridge (1992)
3. Wagner, G., Logic, V.: Knowledge-Based Reasoning with Two Kinds of Negation. Springer, New York (1994)
4. Wagner, G.: Partial Logics with Two Kinds of Negation as a Foundation for Knowledge-based Reasoning. In: Gabbay, D., Wansing, H. (eds.) What Is Negation? Oxford University Press, Oxford (1999)
5. Wagner, G.: Web Rules Need Two Kinds of Negation. In: Bry, F., Henze, N., Małuszyński, J. (eds.) PPSWR 2003. LNCS, vol. 2901, pp. 33–50. Springer, Heidelberg (2003)
6. Analyti, A., Antoniou, G., Damasio, C.V., Wagner, G.: Negation and Negative Information in the W3C Resource Description Framework. Annals of Mathematics, Computing & Teleinformatics (AMCT) 1(2), 25–34 (2004)
7. Kaneiwa, K.: Negations in Description Logic-Contraries, Contradictories, and Subcontraries. In: Dau, F., Mugnier, M.-L., Stumme, G. (eds.) Contributions to ICCS, pp. 66–79 (2005)
8. Ferré, S.: Negation, Opposition, and Possibility in Logical Concept Analysis. In: Missaoui, R., Schmidt, J. (eds.) Formal Concept Analysis. LNCS (LNAI), vol. 3874, pp. 130–145. Springer, Heidelberg (2006)
9. Pan, Z.H., Zhu, W.J.: A New Cognition and Processing on Contradictory Knowledge. In: IEEE Proceedings of 2006 International Conference on Machine Learning and Cybernetics, vol. 1-7, pp. 1532–1537 (2006)
10. Pan, Z.H.: Five Kinds of Contradictory Relations and Opposite Relations in Inconsistent Knowledge. In: Proceedings of IEEE-Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007), vol. 4, pp. 761–764 (2007)
11. Pan, Z.H.: Differentiation and Processing on Contradictory Relation and Opposite Relation in Knowledge. In: Proceedings of IEEE-the 3rd International Conference on Natural Computation (ICNC 2007), vol. 4, pp. 334–338 (2007)
12. Zhu, W.J., Xian, X.: Proposition Calculus System of Medium Logic (I) (in Chinese). Nature Journal 8(4), 315–316 (1985)
13. Xian, X., Zhu, W.: Predicate Calculus System of Medium Logic (I) (in Chinese). Nature Journal 8(7), 540–542 (1985)
14. Zhu, W.J., Xian, X.: On the Mathematical Models of Medium Mathematical System MM^* . Journal of Mathematics Research & Exposition 1, 139–151 (1988)
15. Pan, Z.H.: Construction of a Model of Medium Logical Calculus System (ML). In: Proc. of Workshop on Knowledge-Based Systems and Models of Logical Reasoning, pp. 167–175. Cairo University Press, Cairo (1988)

16. Li, X., Li, G.Y.: The Characteristic Problems of Medium Logic. Chinese Science Bulletin (in Chinese) 33(22), 1686–1689 (1988)
17. Pan, Z.H., Zhu, W.J.: A Finite and Infinite-valued Model of the Medium Proposition Logic. In: Proc. of the Second Asian Workshop on Foundations of Software, pp. 103–106. Southeast University Press, Nanjing (2003)
18. Pan, Z.H., Zhu, W.J.: An Interpretation of Infinite Valued for Medium Proposition Logic. In: Proceedings of IEEE-Third International Conference on Machine Learning and Cybernetics, Shanghai, China, pp. 2495–2499 (2004)

An On-Line Semantic Visualization for Heterogeneous Product Information of the Virtual Organization*

Chengfeng Jian

Software College at ZheJiang University of Technology, HangZhou, P.R. China, 310023
Jiancf@zjut.edu.cn

Abstract. The Semantic annotation is becoming increasingly important for the on-line semantic reconstruction of the heterogeneous product information in the virtual organization (VO) with the semantic grid. Aim at the uniform knowledge representation on the heterogeneous product information of VO, the XOEM+OWL is put forward. And then the XOEM+OWL based production semantic annotation visualization, named a Linked-Entity Visualization (LEV), is introduced. It uses AJAX/SVG to realize the visualization in the display space of the browser. The implementation of this system is shown at last.

1 Introduction

Because of the characteristics of the autonomy and heterogeneity among the distributed virtual enterprises, the sharing and exchange of product semantic information should meet the requirement of dynamic and agility over the Internet. Currently, there are many research methods to involve in [1,2,3,4,5]. The uniform product knowledge representation on the multi Heterogeneous Product information is the key question. On-line semantic reconstruction over the Internet maybe is the effective way to do. Semantic annotation is becoming increasingly important for the on-line semantic reconstruction of the heterogeneous product information in the virtual organization with the semantic grid [6,7].

In this paper, we put forward a Linked-Entity Visualization System (LEV) for collaborative production semantic annotation which uses the AJAX/SVG technology to display on the browser and bases on the XOEM+OEM semantic model. In order to realize the semantic visualization, at first, XOEM+OWL is put forward, which is the semantic model faced on the uniform product knowledge representation on the multi heterogeneous product information. The mapping STEP Schema Graph and OWL Schema Graph are build as Cos(sc,oc),so we can get the semantic pattern matching degree for the semantic representation on the product information. And then the correspondent LEV data schema is put forward which is based on the XOEM+OWL model. At last the implementation of the LEV system is introduced and the example is presented.

* Supported by the National Natural Science Foundation of China(No. 60603087), the Project of the Science and Technology Department of Zhejiang Province(No. 2007C31006).

2 XOEM+OWL Model

XOEM [8,9] is the data model of the XML-based STEP representation. It is difficult to realize the direct mapping between XOEM and OWL because OWL belongs to the semantic layer and the XOEM belongs to the data layer. XOEM has strong capability on the description of data object but the weak capability on the reasoning of constraint. So it is necessary to build the model that it can realize the conversion from XOEM and introduced from OWL pattern graph[10]. That's called XOEM+OWL.

In the XOEM+OWL, the Schema Graph has two layers: data layer and semantic layer. In the data layer, it converted from XML-based STEP representation that uses XOEM model. We call STEP Schema Graph. In the semantic layer, it is introduced from OWL. Please note it is not the conversion from OWL. It adds the semantic annotation by means of OWL. We call OWL Schema Graph. So the mapping between STEP Schema Graph and OWL Schema Graph should be build in the XOEM+OWL model. Fig.1 shows the XOEM+OWL.

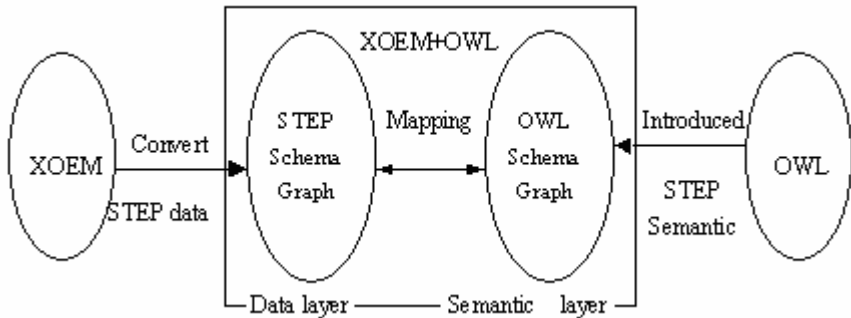


Fig. 1. XOEM+OWL

2.1 The Schema Graph of XOEM+OWL

XOEM+OWL is based on the XOEM model. We can also get the follow definition reference to XOEM:

- Object: = Atomic Object | Complex Object
- Atomic Object: = (oid, label_name, attribute_type, attribute_value)
- Complex Object: = (oid, label_name, Reference)
- Reference := (link, oid, label_name)

Definition 1

Given directed graph $G=(V,E)$.

Assumption: $v_0, v_1 \dots v_i, \dots v_n \in V, e_1, e_2 \dots e_n \in E$.

Exists $r: d \langle r, v_i \rangle > 0, r \in V, v_i \in (V - \{r\}), i=0,1, \dots n$.

Definition 2. Given directed graph $G(V,E,r)$. Exist $G(V_i, E_i), v_i \in V$.

$$V^i = \{v_j \mid v_j \in V \wedge v_k, v_j\};$$

$$E^i = \{<v_j, v_k > \mid v_j \in V^i \wedge v_k \in V^i \wedge <v_j, v_k > \in E\};$$

Rule 1

For the XOME+OWL object, the Node of the directed graph is represented as Object. It is mapping into the Class of OWL.

Rule 2

For the XOME+OWL object’s property, the Edge of the directed graph is represented as Property. It is correspond to the property of the Class or the “hasSubClass” among the classes in the OWL.

The follows description shows the different corresponding relation on the two layers.

2.2 The Data Mapping between STEP Schema and OWL Schema

Every concept from STEP SchemaGraph is compared against concepts from the OWL SchemaGraph. The STEP Schema Graph is used in the data layer on the XOEM+OWL. According to the XOEM, we can get the XML-based STEP representation. So it is necessary to convert from XOEM to XOEM+OWL. The OWL Schema Graph is used in the semantic layer on the XOEM+OWL. The semantic representation of the XOEM+OWL is introduced from OWL. We can build the semantic annotation on the XOEM+OWL. The function listed in Table1 calculates the match score (Cos) between a STEP SchemaGraph concept and OWL SchemaGraph.

Table 1. Function on the mapping

Function	Mapping
Inputs	sc, oc \in W mi = (sci, ocj, Cos)
Output	where, Cos is the Match degree calculated for the mapping sci and ocj (Cos \in [0,1])

The Cos(sc,oc) [13] is composed of two different measures Element Level Match (ElementMatch) and Semantic level match (OWLMatch). ElememntMatch provides the linguistic similarity of two concepts whereas OWLMatch takes care of semantic structural similarity. The Cos(sc,oc) is calculated as the weighted average of ElementMatch and OWLMatch as shown in Formula (1).

$$Cos(sc, oc) = \frac{w_{sc} * ElementMatch + w_{oc} * OWLMatch}{w_{sc} + w_{oc}} \tag{1}$$

where , $(0 \leq w_{sc} \leq 1)$ $(0 \leq w_{oc} \leq 1)$

Weights w_{sc} and w_{oc} indicate the contribution of Element level match and Semantic level match respectively in the total match score. If two concepts have matching structure then it is more appropriate to give more weights to OWLMatch. Based on these conditions the values of w_{sc} and w_{oc} are changed as shown in Table 2.

Table 2. the values of w_{sc} and w_{oc}

w_{sc}	w_{oc}	Condition
1	0	Only STEP syntax level representation
0.1	0.9	OWLMatch > 0.9, ElementMatch < 0.9
0.2	0.8	OWLMatch > 0.75, ElementMatch < 0.75
0.3	0.7	OWLMatch > 0.65, ElementMatch < 0.65
0.5	0.5	OWLMatch < 0.5, ElementMatch < 0.5 and XML-based STEP syntax level representation
0	1	All STEP semantic level representation

3 LEV DATA Schema

LEV Data Schema is build for the LEV system display in the browser according to the XOEM+OWL semantic model. It is driven by the semantic data and its data on the relationships between individual entities on the semantic data and the properties of these entities based on the XOEM+OWL semantic model.

3.1 LEV Data Hierarchy and Definitions

A Strategy Map consists of seven Activities. These Activities can be broken down infinitely into various Entities that exist for each Activity. For a visual depiction of the seven generic Activities of each Strategy Map was simply identify each Activity and the metric (e.g. cost/value) used to measure the Activity or 1/1 indicating there is no metric.

An Activity

An Activity may be broken down infinitely into particular Entities within that Activity in a project. The data associated with each Activity will be drawn from the XOEM+OWL data and mapped into an XML for visualization. The Activities may be opened in a visual space to depict the underlying collection of Entities which an end-user may use for analysis.

An Entity may be a generic text box, or may represent more specific data such as a product, part, place, person or thing. In the Strategy Map, Entities may be created within each Activity, moved between Activities, cloned, deleted, edited and links between Entities may be specified.

A Group of Entities

A group of Entities reflects subcomponents of an Activity(section) within a Strategy Map.(A specific example could be; the Activity is 'Research' and the Entities are

Library Services, Research Functions, and other various components of the Research Activity).

Any two Entities may have a mathematical relationship with other Entities through a pre-defined data schema. For example if two Entities contain information such as the names of people involved with an operation, a link can be specified based on their joint involvement.

3.2 LEV Data Schema

Table 3. the LEV data schema

LEV DATA SCHEMA	XOEM+OWL
Entity (entity element)	Node
Entity with Property (property-name element and propertyvalue element)	Node with edge joining it to the class with name "hasProperty"
Entity Link (entitylink element)	Edge
Entity Link between entities (entitykey1 to entitykey2)	Edge between the two class nodes
Entity Link with Relationship expression (relationship element)	Edge with operation
Entity Link with the Element match value(strength element)	W_{sc} (the Element match value weights)
Entity Link with the OWL match value(Color element)	W_{oc} (the OWL match weights value)
Entity with the Stage of Activity (stage element)	Node with edge joining it to the class with name "hasInstance"
Entity Link with the start entity expand (lbase element)	Edge between class node to subclass node with name "hasSubClass"

4 Examples

Example: There is a circle named CIRCLE, its center point is circle(50,45) and its radius named Radius is 100mm. For the same STEP example, we can get the two schema graphs according to the XOEM+OWL. One is STEP Schema Graph (Fig.2) corresponds with XML schema and the other is the OWL Schema Graph (Fig. 3) corresponds with the OWL. It can be drawn the conclusion that the two schema graphs are corresponding except the pattern matching degree. It proves the XOEM+OWL can realize the uniform semantic representation and shows the pattern matching degree by means of the Cos(sc,oc). At last, we can get the correspondent LEV data schema(Fig. 4) according to the consistent XOEM+OWL graph so as to get the SVG display on the browser.

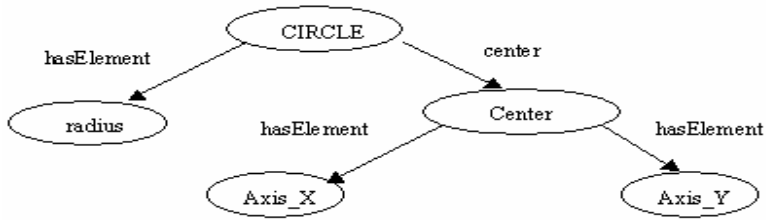


Fig. 2. STEP Schema Graph: the XOEM+OWL schema graph corresponding the XML schema

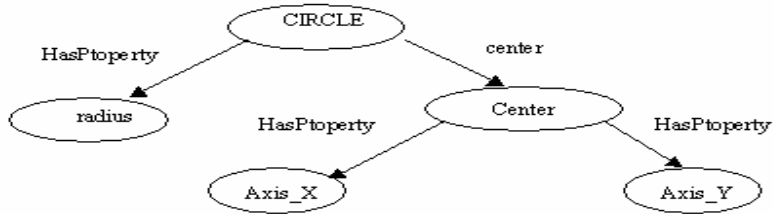


Fig. 3. OWL Schema Graph: the XOEM+OWL schema graph corresponding the OWL

```

- <data>
  <entitystart id="EK_0_circle">circle</entitystart>
  - <entity id="EK_0_circle">
    <entitytype>graph</entitytype>
    <entityname>circle</entityname>
    - <property id="PK_01">
      <propertyname>radius</propertyname>
      <propertyvalue>100mm</propertyvalue>
    </property>
    - <property id="PK_02">
      <propertyname>HAS</propertyname>
      <propertyvalue>center</propertyvalue>
    </property>
  </entity>
  - <entity id="EK_1_center">
    <entitytype>location</entitytype>
    <entityname>center</entityname>
    - <property id="PK_10">
      <propertyname>Axis_X</propertyname>
      <propertyvalue>50mm</propertyvalue>
    </property>
    - <property id="PK_11">
      <propertyname>Axis_Y</propertyname>
      <propertyvalue>45mm</propertyvalue>
    </property>
  </entity>
  - <entitylink id="LK_0_1">
    <entitykey1>EK_0_circle</entitykey1>
    <entitykey2>EK_1_center</entitykey2>
    <propertykey1>PK_02</propertykey1>
    <property>HAS</property>
    <lbasevalue>center</lbasevalue>
    <strength>5</strength>
    <color>CC8F33</color>
  </entitylink>
</data>
    
```

Fig. 4. LEV data schema

5 AJAX/SVG Based LEV System

According to the LEV data schema, it will pop up a new window that is initially populated with a glyph representing that Entity. The key function which must be

provided by the Linked-Entity Visualization (LEV) is the ability to visually manipulate many Entities in the display space of a browser and show connections between entities based on their properties.

In order to provide the user with an experience that allows them to work with several hundred Entities of a project in the same way that a CAD Designer works with

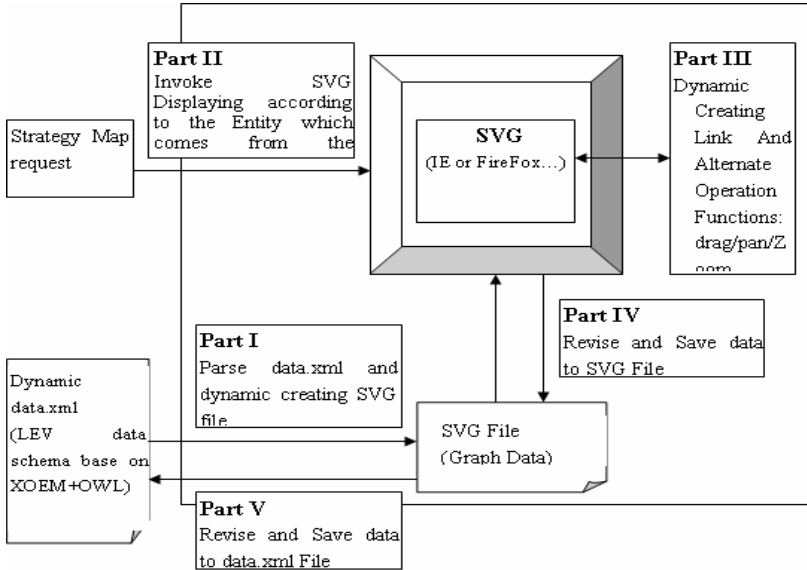


Fig. 5. LEV system principle

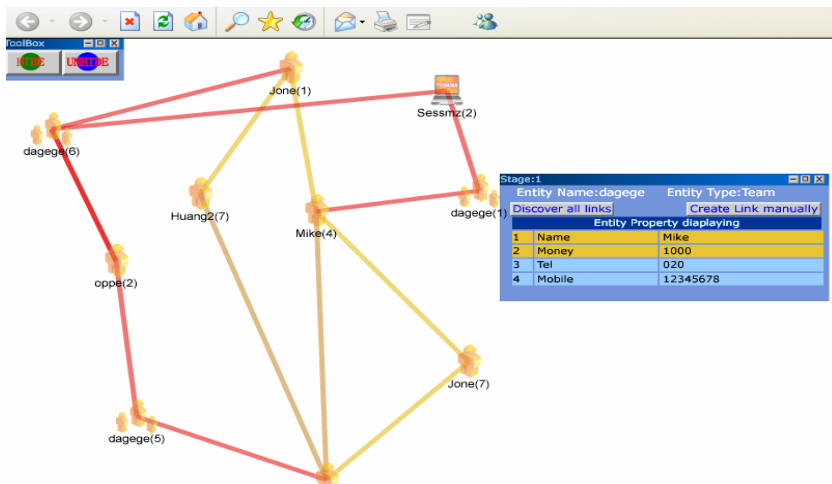


Fig. 6. LEV system implementation

technical drawings of a tooling system, The users can have the ability to create, clone, delete, edit, move, and link Entities in groups, such that information can be added or changed within a project map as easily possible.

LEV can show logical connection between Entities based on the LEV data schema which contains all of the links between Entities based on all of their properties. Users may manipulate data in fields themselves, and also change values by manipulation in the visual space.

Fig.5 shows the realization of the LEV system. Fig.6 is an example of this system implementation.

References

1. Lilia, G., Dominique, R., Michel, T.: Product Information Systems Engineering: an Approach for Building Product Models by Reuse of Patterns. *Robotics and Computer-Integrated Manufacturing* 19(3), 239–261 (2003)
2. Advanced Technology Program NIST, <http://www.atp.nist.gov/>
3. Eurostep - Information Solutions for a Global Age, <http://62.181.193.208/>
4. West, M.: An Overview of the Modularization, SC4 Data Architecture Projects [EB/OL] (2004), <http://www.nist.gov/sc4/wg-qc/wg10/current/n291/wg10n291.pdf>
5. Metzger, F.: The Challenge of Capturing the Semantics of STEP Data Models Precisely [EB/OL] (2004), <http://www.ladseb.pd.cnr.it/infor/Ontology/Baselpapers/Metzger.pdf>
6. Ludwig, S.A., Reyhani, S.M.S.: Semantic Approach to Service Discovery in a Grid Environment. *Web Semantic: Science, Services and Agents on the World Wide Web* 4, 1–13 (2006)
7. Uren, V., Cimiano, P., Iria, J.: Semantic Annotation for Knowledge Management: Requirements and a survey of the state of the art. *Web Semantic: Science, Services and Agents on the World Wide Web* 4, 14–28 (2006)
8. Jian, C.F., Tan, J.R.: Description and Identification of XML-based STEP Production Data. *Journal of Computer-Aided Design & Computer Graphics* 13(11), 983–990 (2001)
9. Jian, C.F., Tan, J.R.: The Uniform Information Representation for On-line Reconstruction of Virtual Enterprise Base On XML. In: *Proceedings of the Third International Asia-Pacific Web Conference, Xi'an, China*, pp. 292–293 (2008)
10. Ankolekar, A., Burstein, M., Hobbs, J., Lassila, O., Martin, D., McDermott, D., McIlraith, S., Narayanan, S., Paolucci, M., Payne, T., Sycara, K.: DAML-S: Web service Description for the Semantic Web. In: *Proceedings of the 1st International Semantic Web Conference* (2002)

An Optimized Parallel Decision Tree Model Based on Rough Set Theory

Xiaowang Ye and Zhijing Liu

School of Computer Science and Technology
Xidian University, Xi'an 710071, China
yexiaowang@gmail.com, liuprofessor@163.com

Abstract. This paper presents an optimized parallel decision tree model based on rough set theory, first the model divides global database into subsets, then using the intuitive classification ability of decision tree to learn the rules in each subset, at last merge each subset's rule set to obtain the global rule set. In this model, with the uncertain information analysis method of rough set, the author presents a massive data segmentation method and using the Weighted Mean Roughness as a decision tree method for attribute selection. This parallel data mining model with the best segmentation algorithm based on rough set can be well used in dealing with massive database.

Keywords: Parallel Decision Tree; Rough Set theory; Data Mining.

1 Introduction

The Rough Sets theory is a mathematical method used to deal with imprecise and uncertain information; it was proposed by Pawlak Z. Poland scientists in 1982. At present, data mining technology research mainly concentrated in the classification, clustering, association rules and so on, which classified data mining is the most active and most sophisticated research direction. So, the classification algorithm is become a very important study. Decision trees method is one of the most effective classification rules learning algorithm and one of the most widely used in inductive inference algorithm in recent years. The traditional random sampling method and random segmentation method is not ideal in stability of correct rate. Compared with random sampling, the classical knowledge gain algorithm although has improved its correct rate, it takes more time in processing than random sampling. That means it takes more time in order to improve accuracy. So, this paper presents a new parallel Decision Trees Model based on rough sets theory, with the high correct rate, is greatly improved the efficiency of handling massive data mining.

2 Basic Concepts of Rough Sets and Decision Trees

2.1 Rough Set of the Related Concepts

2.1.1 Knowledge Representation and Decision Making System

Definition 1. $S=(U,A,V_{a,f})$ is defined as a knowledge Representation system, suppose U is a non-empty limited sets, called domain; suppose A is a non-empty limited sets,

called attribute sets; V_a is the range of attribute $a \in A$; $a: U \rightarrow V_a$ defined as single alluding, for any one attribute element of U sets ,there is only one value corresponding in V_a sets. If A is composed by condition attributes sets C and conclusion attributes sets D , meet $C \cup D = A, C \cap D = \emptyset$, then S is called a decision-making system. Order to express simple, sometimes with $(U, C \cup \{d\})$ to express a decision-making system., that is conclusion attributes sets contain one element.

Rough sets theory introduces the concepts of approximate sets. The concept of X in U , under the equivalence relations R , then $R_X = \cup \{Y: Y \in U/R, Y \subseteq X\}$
 $R^-X = \cup \{Y: Y \in U/R, Y \cap X = \emptyset\}$. [1]

Definition 2. For two equivalence relation

$P, Q, P \subseteq C, Q \subseteq D$. Q 's positive domain in p is defined as : $pos_p(Q) = \cup P_X(X \in U/Q)$. For equivalence relation $R \in P$, if $pos_p(Q) = pos_{p_{\{R\}}}(Q)$ called R is unnecessary for Q . The all necessary equivalence relation set in P is the core Q of P , denoted by $core_p(Q)$. [1]

Definition 3. Indiscernibility relation

For decision system $S=(U,A,V_a,f)$, $\forall E \subseteq A$ is a subset of condition attribute set, called binary relation $Ind(E)$ is the indiscernibility relation of S ;
 $Ind(E) = \{(x,y) \in U \times U | \forall a \in E, f(x,a) = f(y,a)\}$ represents object x and y is indiscernibility relation about the subset E of attribute set A . [2]

Definition 4. Upper approximation and Lower approximation

For information system $S=(U,A,V_a,f)$, let $E \subseteq A, X \subseteq U$, then $Ind(E)$ is equivalence relation on $U \times U, E(X_i)$ is the equivalence class including X_i according to equivalence relation $Ind(E)$, called $E(X_i)$ is the basic set of E . U divisions by E attribute, it will obtain a equivalence class set. The upper approximation $B_+(X)$ and lower approximation $B_-(X)$ of subset X is defined as follows:

$$B_+(X) = \{xi \in U | B(xi) \subseteq X\} \quad B_-(X) = \{xi \in U | B(xi) \cap X = \emptyset\}$$

2.2 Decision Trees Concept and Principle

Decision Trees category examples by array the examples from the root to leaves, leaf nodes are the classification of examples. Each node corresponds to the examples of the attributes test, and each node corresponds to the subsequent branch attributes of a possible value. Category examples is begin with root node of the tree, tests the specified attributes, then follow the examples given in the attributes value corresponding branches moving down. This process is executed again in a new root node of the tree. How to select attributes for test is the key of decision trees algorithm. Majority decision trees learning algorithm has been developed is based on the core algorithm ID3 and subsequent C4.5 algorithm. Use the greed of top-down search method to traverse the possible decision tree space. Algorithm C4.5 takes the largest information gain rate as the splitting attribute. In this model, it takes weighted mean roughness as the attribute selection standard [3]. Compared with C4.5, it can reduced the complexity of decision tree. The decision tree obtained by this method is simpler than C4.5, occupies less spatial, has fewer leafy nodes and more easily to make

decision. It is also avoided testing certain attribute for many times in one path of decision tree.

Definition 5. Weighted Mean Roughness [7]

For decision system $S=(U,A,V_a,f)$, $X \subseteq U$, $E \subseteq A$,

$\Upsilon_B(X)=\text{card}(B_{-}(X))/\text{card}(B(X))$ is the accuracy in space E , $0 \leq \Upsilon_B(X) \leq 1$, the Weighted Mean Roughness of X about E is defined as:

$$\Gamma_E(i) = 1 - \left(\sum_{j=1}^m w_j \Upsilon_E(X_j) \right)$$

i is the i th condition attribute; j is the j th equivalence class of decision attribute, $j=1\dots m$, m is the equivalence class number of decision attribute; X_j is the j th equivalence class set of decision attribute, $X_j \subset U$; ω_j is the proportion of X_j in universe U : $\omega_j = \text{card}(X_j) / \text{card}(U)$ The value range of $\Gamma_E(i)$ is $[0,1]$, if $\Gamma_E(i)=0$, that is a determined division on i th attribute, there is no uncertain factors; if $\Gamma_E(i)=1$, that is the largest uncertain division on i th attribute. So, $\Gamma_E(i)$ is smaller, that is more determined division on i th attribute. The weighted value here is to consider a more comprehensive contribution on each division of decision attribute.

3 Massive Data Mining Algorithm and Parallel Mode

Usually, the classic distributed parallel data mining algorithm using the same serial data mining algorithms for distributed data sets to obtain the local node model, then combination all local models for the final model. In this paper,, the author takes the parallel measures in order to increase efficiency and to deal effectively with the knowledge of data mining. First, it takes the optimal division based in rough sets for massive data sets to obtain a number of subsets. Second, using the parallel algorithm based on rough sets and decision trees to learn the subsets on different processors. Generate each subset decision tree model. Finally using certain rules to integrated local modules into the overall final module (Fig. 1).

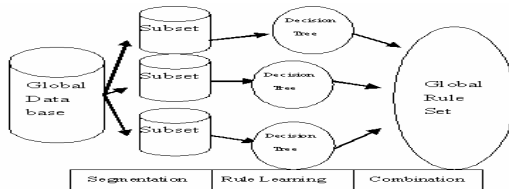


Fig. 1. Parallel Data Mining learning process

3.1 Massive Data Segmentation Based on the Rough Set

Segmentation is the core of massive data mining, In the entire process of data segmentation, it is need to maintain the original data sets classification capacity unchanged. Therefore, in the division, how to make the important information

retained and subsets load balancing is the main consideration. For an information system, if the knowledge or rules from division information system equal to original information system, we can say such a separation is the best division.

Definition 6. Given information system $S=(U,C \cup D,V,f)$, S through a segmentation method cut into k sub-systems: $S_1,S_2,\dots,S_j,\dots,S_k$. $S_j=(U_j,C \cup D,V_j,f_j)$, $U=U_1 \cup U_2 \cup \dots \cup U_k$ If $(\forall P \subseteq C)[pos_p(D)]_S = \bigcup_{j=1}^k pos_p(D)_S$

$pos_p(D)_S$ is the P positive domain of D in the definition of information system S_j , then we call this segmentation is the best one[2]. Segmentation must meet to two main conditions: the all samples of compatible classification X have the same attributes value and decision values in A , these samples are also same as in combination of attributes A . So these samples can be arbitrarily divided into different subsets, will not change its positive domain. The conflict conditions of samples have same attributes value but different decision-making values, arbitrarily divided it will lead to increase compatibility conditions. So, we assigned the samples of conflict classification Y to the same data set. For some continuity attributes, we can preprocess the data and discrete. For some attributes missing, the first we should do is data integrity processing. As achieve the best segmentation require all combination of conditions attributes, this is not very realistic in general. And because of the relevancy among condition attributes, we just consider only part of the condition attributes combinations and give up others in general. Of course, this will lead to the lower correct rate. So this paper presents a segmentation algorithm based on rough sets for the actual application. The experimental tests show that it can be very well in keeping important information and reducing the segmentation time. Algorithm's main steps are as follows:

Input: Global data sets (S), segmentation numbers (m);

Output: data sets S_1, S_2, \dots, S_m .

Step1: For each attribute A_i , do calculate the importance of this attribute.

Step2: C is the attributes sets initialized by ascending importance of all condition attributes, D is the decision attribute, S_k is the k th subset ($1 \leq k \leq m$), $|S_k|$ is the sample numbers in S_k , $M=|S|/m$ (M express the average sample number of m data sets);

Step3: while attributes non-empty do{

Search all the conflict condition classifications in attributes combination C .

Ascending all the searched conflict condition classifications by the number of samples contained, be: X_1, X_2, \dots, X_n , $1, 2, \dots, n$ is the conflict condition classification number, $|X_j|$ express the sample numbers in the j th conflict condition classification. There $|X_1| \geq |X_2| \geq \dots \geq |X_n|$; For every conflict classification X_j ($1 \leq j \leq n$) do { If conflict condition classification numbers $|X_j| \geq m$, then $p=|X_j|/m$, $q=\text{Mod}(|X_j|/m)$ for each

subset S_k , insert p samples into conflict classification number j , if $q \neq 0$, then ascend the sample numbers of subsets $|S_1| \leq |S_2| \leq \dots \leq |S_m|$, then insert q samples which conflict classification is j into the first q subsets. Else ascend subsets by each sample numbers. $|S_1| \leq |S_2| \leq \dots \leq |S_m|$, insert the p samples into first p subsets, then remove X_j which conflict classification is number j from S . } If S is NULL, then exit for, to the forth step.}

Step 4: Algorithm end.}

3.2 Decision Tree Algorithm

This decision tree algorithm is the weighted mean roughness decision tree algorithm [5]. It is used weighted mean roughness as its attribute partition standard. It takes the minimum weighted mean roughness as its best splitting attribute, then uses the selected attribute to divide the training set. In the division of training set, it takes the same method to select the best attribute and division.

The main steps of decision tree construction are as follows:

Input: subset S_j ($1 \leq i \leq m$); attribute set A_i ($1 \leq i \leq m$)

Output: decision tree;

Step 1: According to condition attribute in the subset S_i , calculate each attribute's upper approximation and lower approximation relative to decision attribute. Then calculate the attribute's weighted mean roughness.

Step 2: Choose the attribute which has the smallest weighted mean roughness for the first splitting attribute T ;

Step 3: Construct the decision tree with the root of T , Root(T) (Root express the root node) for each possible value V_j of T , under root T add a new branch, test the T 's value is V_j , X_{vi} is the subset meeting T 's value is V_j in Q , for each subset, if not reach the leaf node, then return step 2;

Step 4: if all attributes are the leaf nodes, then end.

3.3 Rule Sets Mergence

Rule sets mergence takes simple integration solutions. Original data set S is divided into m subsets S_1, S_2, \dots, S_m . $|S|$ is the sample numbers of data set S . R_1, R_2, \dots, R_m , are the rule sets of each subset, the main steps of Rule sets mergence are as follows:

Calculate the confidence level conf and support level sup of each subset, then for each subset, obtain the weighted value of conf and sup in global dataset S : $\text{conf} \times |S_j| / |S|$, $\text{sup} \times |S_j| / |S|$;

Merge each sub rule set, combined with the same rules as a rule, then merge its conf and sup of each same rule, the result as the final conf and sup in global dataset.

4 Illustrative Examples and Interprets Significance

In the experiment with this parallel data mining model system PDM, the author realized decision tree learning based on the above model system, and uses 4 databases from the UCI machine learning database which shown in Table 1 for the accurate test and superior test.

Table 1. 4 databases is selected from UCI

Datasets	ECOLI	CAR	MONK	ABALONE
Samples	336	1728	432	4177
Attributes	8	6	7	8

In order to compare with the capability, we use 3 methods for obtaining rules.

Method 1: It takes the classical knowledge gain algorithm based on rough set, combine with the data mining model shown in figure 1 to obtain the rules.

Method 2: It takes segmentation algorithm based on rough set proposed by this paper, combine with the data mining model shown in figure 1 to obtain the rules.

Method 3: It takes Random sampling method to obtain subsets, and then takes decision tree learning to obtain the rules.

For each dataset, sample number is n, the subset number is m, takes the most first match strategies for test. Fig 2-3 show the 4 databases’ test result with 3methods above. CRseg ,CR ,CRsam represent separately the accuracy of method 1,2,3.

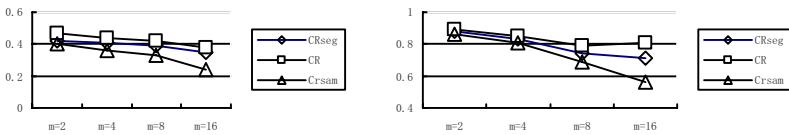


Fig. 2. Test result of accuracy with ECOLI and CAR

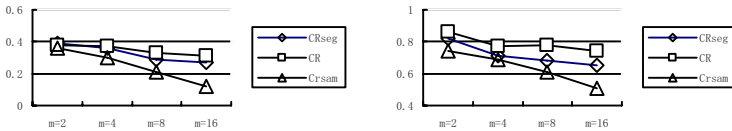


Fig. 3. Test result of accuracy with MONK and ABALONE

The test results show that these three methods can obtain the quite good effect when m is less. This is because the subsets have quite many samples when m is less, the method 1 and method 3 can be very good response to the implied knowledge in

global datasets. Along with m increasing, CR_{seg} and CR_{sam} have lower accuracy. The random sampling has reduced the most quickly, which shows that random sampling has lost more and more important information when m is increasing, and to the influence to global rule set is gradually increasing. The classical knowledge collection of rough set can obtain more correct rate stability than random sampling, but with the increasing in m also has a certain decline. With m increasing, although the correct rate of CR has a little reduced, the overall stability is stronger than the result of method 1 and method 3. That means can still obtain better result when subset numbers are more.

In addition, we also do the speed test with samples in database ABALONE. Comparing with the data mining times are required by method 1,2,3. TCRseg, TCR, TCRsam represent separately the time of using method 1,2,3, in units of seconds. As shown in table 6: accurate test and superior test.

Table 2. Speed test result with ABALONE

m	TCR_{seg}	TCR	TCR_{sam}
2	108	105	103
4	75	73	72
8	64	62	62
16	58	58	58

From the result in Table 6, we can see that the requiring time by data mining with method 2 is slightly longer than method 3 and slightly shorter than method 1. this mainly because method 2 uses the segmentation based on rough set requiring longer time than random sampling, but comparing with classical knowledge gain algorithm based on rough set, it takes less time on segmentation. Moreover, with the m increasing, the model proposed by this paper can reduce the time of data mining required obviously. Generally speaking, comparing with the classical segmentation method, we can reduce the time and improve the stability of correct rate obviously. From these experiments, we can know that parallel decision tree model based on rough set can be very good at massive data mining.

5 Conclusion

At present, the rough set has become a perfect method in data preprocessing, attributes reduction, rule obtaining and so on in data mining theory. Combine with the rough set’s qualitative analysis and the decision tree’s visual expression can be a satisfactory solution to the problem. This model has not considered multi-attribute division and incremental learning. Therefore, expanding the model to be multivariable incremental learning model is the subject for future research.

Acknowledgments. This research project was incorporated and performed as a part of “Optimizations on Distributed Data-mining” project promoted by National Natural Science Foundation of China (NSFC) No. 60573139.

References

1. Pawlak, Z.: Rough Sets. *International Journal of Computer and Information Sciences* 11, 341–356 (1982)
2. Pawlak, Z.: Rough Fuzzy Hybridization: Rough Functions and Rough Calculus. In: Pal, S.K., Skowron, A. (eds.), pp. 99–109. Springer, Heidelberg (1999)
3. Quinlan, J.R.: *Machine Learning: Induction of Decision Trees*. Morgan Kaufmann, San Mateo (1993)
4. Suresh, K., Choubey, J.S., Deogun, V.V., Raghavan, H.: Server: A Comparison of Feature Selection Algorithm in the Context of Rough classifiers. In: 5th IEEE International Conference on Fuzzy Systems, New Orleans, LA, vol. 2, pp. 1122–1128 (1996)
5. Xi, Z., Wang, J., Sun, H.W., Yang, M., Zhao, H.: A Comparison between Fuzzy and Crisp Decision Tree. *Computer Engineering and Applications* 39, 72–75 (2003)
6. Janikow, C.Z.: Fuzzy: Decision Trees: Issues and Methods. *IEEE Transactions on Systems, Man and Cybernetics* 28(1), 1–14 (1998)

A Topic-Specific Web Crawler with Concept Similarity Context Graph Based on FCA

Yuekui Yang¹, Yajun Du¹, Jingyu Sun², and Yufeng Hai¹

¹ School of Mathematical and Computers Science, Xihua University,
Chengdu 610039, China

yuekuiyang@gmail.com, duyajun@mail.xhu.edu.cn

² College of Computer and Software, Taiyuan University of Technology,
Taiyuan 030024, China

whitesunpersun@163.com

Abstract. With Internet growing exponentially, topic-specific web crawler is becoming more and more popular in the web data mining. How to order the unvisited URLs was studied deeply, we present the notion of concept similarity context graph, and propose a novel approach to topic-specific web crawler, which calculates the unvisited URLs' prediction score by concepts' similarity in Formal Concept Analysis (FCA), while improving the retrieval precision and recall ratio. We firstly build a concept lattice using the visited pages, extract the core concepts which reflect the user's query topic from the concept lattice, and then construct our concept similarity context graph based on the semantic similarities between the core concepts and other concepts.

Keywords: Topic-specific spider; Search engine; Formal concept analysis.

1 Introduction

Today, search engine has becoming an effective tool of finding relevant information from Internet. But with the amount of web sites and documents growing even faster and site contents getting updated more and more often, large-scale search engines cover an even-decreasing segment of the web. The recall ratio of even the large-scale search engines is rather low, covering 50-70% of the web today [1]. Several search engines using topic-specific web crawler have been generated in recent years. The topic-specific web crawler collects web pages from the Internet by choosing to gather only particular pages related to a specific topic and needs small storage. At the same time, they offer higher retrieval precision and recall ratio than large-scale search engines.

The topic-specific crawler works like a spider, it traverses the web graph according to a topic-relevant ordering instead of the breadth-first or depth-first ordering. One of the major problems of the topic-specific crawler is how to assign a proper prediction score to the unvisited URLs. A novel approach called context graph is proposed in [2], which stored the information of link-hierarchies and the distance from general pages to topic web pages. In [3], the authors improve the approach,

called therelevancy context graph, the distance is calculated with the construction of both general word and topic-specific word distributions. But in [2, 3], the nodes in context graph are web pages, and the distance among nodes did not reflect the semantic relevancy. In [4], they develop a latent semantic indexing (LSI) classifier that combines link analysis with text content. A learnable topic-specific web crawler is studied in [5], their method is to concentrate on improving the next crawling process according to an interested topic. Though the crawlers mentioned above use very efficient technologies, no one has mentioned computing prediction score of unvisited URLs based on the concepts’ semantic similarity in Formal Concept Analysis.

Formal Concept Analysis (FCA) [6] is a method for data analysis, knowledge representation and information management, and it have a significant potential for application. Galois (concept) lattice was built up by Wille in 1982, it is the core of the mathematical theory of FCA and is an important data structure of knowledge representation. FCA can help the Natural Language Processing (NLP), because it can extract concepts from the document. Some lattice-based IR systems have been presented such as CREDO [7] and so on.

By above researches, this paper presents a new notion called concept similarity context graph, and proposes a novel topic-specific web crawler. Each of the unvisited URLs is assigned a prediction score for the next traversal.

2 Knowledge Background

2.1 Formal Concept Analysis

Firstly, we recall some basic notions of FCA. The definitions in this subsection are quoted from [6].

Definition 1. A formal context is a triple $K = (G, M, I)$ where G and M are sets and $I \subseteq G \times M$ is a binary relation. The elements of G are called objects and the elements of M are called attributes. The inclusion $(g, m) \in G \times M$ is read “object g has attribute m ”.

In our discussion, $G = \{1, 2, 3, 4, 5, 6\}$ describes web pages, and $M = \{a, b, c, d, e, f\}$ describes the term set of G . $K = (G, M, I)$ is a formal context, and its binary relation described in Table 1.

Definition 2. Given $K = (G, M, I)$ a formal context. For a web page set $X \subseteq G$, a term set Y defined on M , the operators \uparrow and \downarrow are defined as follows:

$$\uparrow : X^\uparrow = \{ m \in M; \forall g \in X, (g, m) \in I \} \tag{1}$$

$$\downarrow : Y^\downarrow = \{ g \in G; \forall m \in Y, (g, m) \in I \} \tag{2}$$

Table 1. The binary relation of the formal context, and ‘×’ presents a page has the corresponding term in the above

term	a	b	c	d	e	f
page						
1	×	×				×
2	×		×	×	×	
3	×	×		×		
4		×	×			×
5	×					×
6		×	×		×	×

Definition 3. A formal concept of a formal context $K = (G, M, I)$ is a pair (X, Y) with $X^\uparrow = Y$ and $Y^\downarrow = X$. The set X is called the formal concept’s extent and the set Y its intent.

Definition 4. The set $\beta(K)$ of all concepts of a formal context K together with the partial order $(A_1, B_1) \leq (A_2, B_2) \Leftrightarrow A_1 \subseteq A_2$ (which is equivalent to $B_1 \supseteq B_2$) is called concept lattice of K .

Fig.1 is the hasse graph of the concept lattice that corresponds to the formal context in Table 1, it includes all the concepts and the partial orders between concepts.

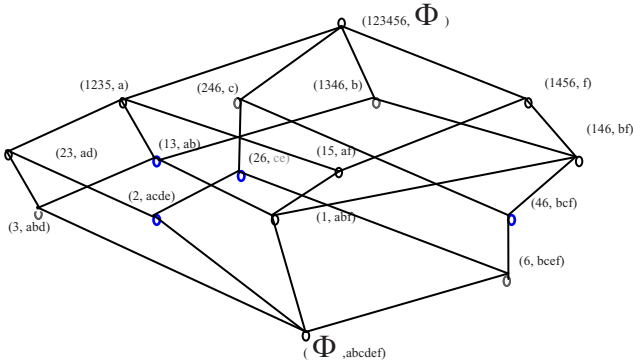


Fig. 1. The hasse graph of the concept lattice that corresponding to the formal context in Table 1. The four blue nodes are the core concepts

2.2 Concept Similarity

In this section the notion of similarity between concepts in FCA is recalled. The similarity of concepts and the matching between lattices are studied in [8, 9]. Our method is similar to the method in [8], which is an ontology-based method for assessing similarity between FCA concepts.

Definition 5. (Term cell) A term cell is a term set in which the terms are synonymic or have the same meaning in a domain.

Definition 6. We define our concept-concept similarity between (E_1, I_1) and (E_2, I_2) as follows:

$$SimCC((E_1, I_1), (E_2, I_2)) = \frac{|(E_1 \cap E_2)|}{r} * w + \frac{|(I_1 \cap I_2)|}{m} * (1 - w) \tag{3}$$

Where r is the greater between the cardinalities of the sets E_1 and E_2 , m is the greater between the term cells of the sets I_1 and I_2 , w is a weight, such that $0 \leq w \leq 1$, we define $w = 0.5$ in this paper. $E_1 \cap E_2$ presents how many the same elements between E_1 and E_2 , while $I_1 \cap I_2$ presents how many the same term cells between I_1 and I_2 .

Example 1: (26, ce) as C_1 and (2, acde) as C_2 are two concepts of the concept lattice in Fig.1, we assume the elements a and c is considered a term set, and then $r=2, m=3$ here and $SimCC(C_1, C_2) = \frac{1}{2} * 0.5 + \frac{2}{3} * 0.5 \approx 0.58$.

2.3 Core Concept

We introduced the core concept of a formal context for construction our concept similarity context graph by calculating similarity between core concepts and general concepts, while the core concepts would specifically reflect the user’s query topic.

Definition 7. (Core concept) Given a term set QE and a concept (A, B) , if $\exists Q$, where $Q \subseteq B$ and not exist a concept (C, D) while $(C, D) \leq (A, B)$ and $Q \subseteq D$, then the concept (A, B) is called a core concept.

Where QE is the expansion term set of the query set inputted by users using our term cells, Q is a element of QE , (A, B) is a concept of a formal context.

Example 2: Given a query set $\{c \wedge e\}$ by a user, then we expand the query set using the term cells, assuming $(a \vee c)$ and $(b \vee e)$ are two term cells, then the expansion term set is $\{(a \vee c) \wedge (b \vee e)\}$, i.e. $\{\{a \wedge b\} \vee \{a \wedge e\} \vee \{c \wedge b\} \vee \{c \wedge e\}\}$. According to definition 10, the four concept (26,ce), (46,bcf), (13,ab) and (2,acde) are the core concepts of the concept lattice in Fig. 1.

3 The Topic-Specific Crawler with Concept Similarity Context Graph

In this section, we illustrate our topic-specific crawler specifically. As Fig. 2 suggests, the system architecture includes several parts, such as query expansion, constructing concept lattice, extracting the core concepts, building concept similarity context graph

based on concepts' similarity, calculating the unvisited URLs' prediction scores, and so on.

We mainly discuss two parts, one is how to build the concept similarity context graph based on concepts' similarity in Formal concept analysis, and the other is to calculate the unvisited URLs' prediction scores in our concept similarity context graph. In the other parts of our system, we find some synonymic term sets as the term cells and use the method in [10] to construct the concept lattice. In the following subsections we will study the two main parts respectively.

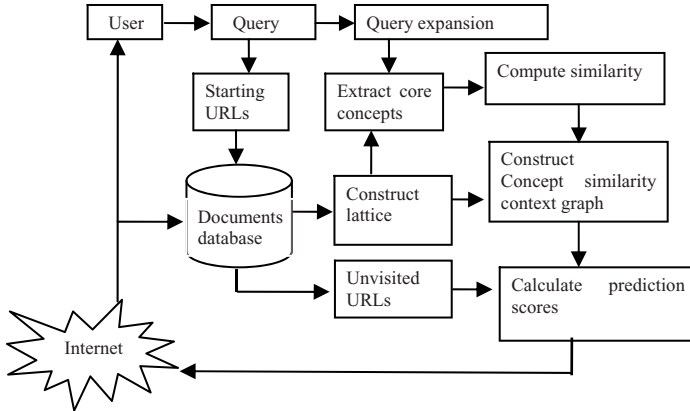


Fig. 2. The architecture of our topic-specific crawler with concept similarity context graph

3.1 Concept Similarity Context Graph

Definition 8. (The score of the concept in the concept lattice) Each of the concept c 's score reflects the semantic similarity between itself and the core concepts. It is defined as follows:

$$Score(c) = \begin{cases} 1 & \text{when } c \text{ is a core concept} \\ Score(d) * SimCC(c, d) & \text{when } c \text{ is not a core concept} \end{cases} \quad (4)$$

Where d is the concept c 's parent-concept or child-concept, and $SimCC(c, d)$ is the similarity between c and d in the concept lattice, $Score(d)$ is the concept d 's score.

Definition 9. (Concept similarity context graph) A concept similarity context graph is a graph which corresponds to a concept lattice, a node in the graph is a concept in the concept lattice and a side in the graph reflects one node is the other node's parent-concept or chilled-concept in the concept lattice. While the graph has to satisfy the following two conditions:

1. The core concepts must be in the middle of the graph.
2. The score of each node in the graph have to be the largest according to Definition 8, which is called the maximum score rule.

We have constructed a concept lattice from the formal context of our visited pages. The Fig. 3 describes the concept similarity context graph. And how to build the concept similarity context graph using the above concept lattice is discussed in this subsection. Our method is divided to four steps as follows:

- a) Firstly, we find the core concepts from the concept lattice, then insert these concepts to the middle of our concept similarity context graph, because these concepts has the largest similarity.
- b) When the core concepts' parent-concepts or their child-concepts are not core concepts, we insert them in a set A, while the concepts' score is calculated.
- c) We select the concept of the largest score and insert the concept to the concept similarity context graph. If the concept' parent-concept or child-concept is not included the concept similarity context graph, then insert the concepts to the set A while calculating the concepts' score; else we update the concepts' score in the set A according to the maximum score rule.
- d) The fourth step is to repeat the third procedure until all the concepts in the concept lattice are inserted the concept similarity context graph.

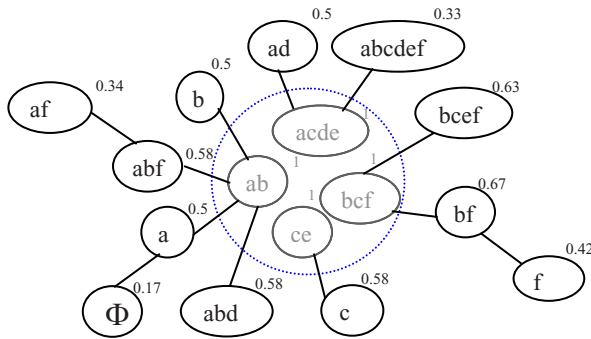


Fig. 3. The concept similarity context graph corresponding to the concept lattice in Fig. 1, each node is a concept which is represented by its intent, the number in the top right corner is each node's score

3.2 The URLs Prediction Scores

How to evaluate the unvisited URLs plays a key role in the process of topic-specific web crawling. If we can arrange the proper order efficiently, the web crawler could collect more topic-specific web pages while traversing less off-topic web pages. In the above subsection, a concept similarity context graph has been built based on the concept lattice.

Definition 10. (Prediction score) A unvisited URL as p , its prediction score is calculated by these concepts' score, the extents of these concepts include p 's parent web page, as follows:

$$Pr\ score(p) = \frac{\sum_{i=1}^n Score(C_i)}{n} \tag{5}$$

Where C_i is a concept and its extent includes the p’s parent web page, n is the total number of the concepts which include the p’s parent web page, such as C_i .

4 Experiments

We download about 5000 web pages form the yahoo sport directory, and record their link relationship as our experiment’s data set. For evaluating our method, we compare our method with other two crawlers, one is the breadth-first crawler, the other is a focused crawler using cosine similarity. We query the topic about “basketball” in the data set, Fig. 4 demonstrates our method is more ascendant than other two crawlers in retrieval precision.

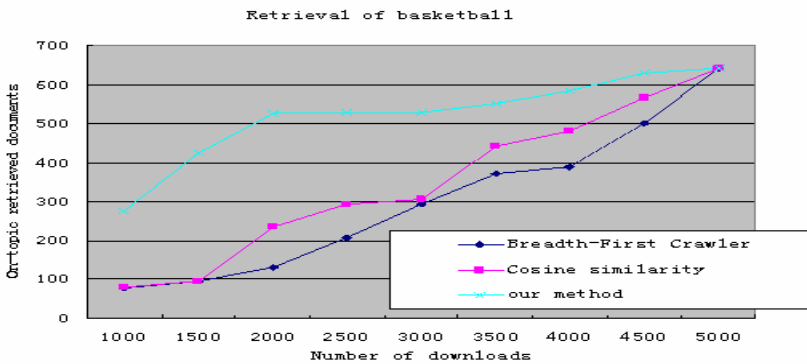


Fig. 4. The on-topic retrieved documents about the topic “basketball”, which reflect our method could retrieve more relevant documents in the case of retrieving the same quantity of documents

5 Conclusion and Future work

In this paper, we present the notion of concept similarity context graph, and propose a novel approach to topic-specific web crawler which calculates the unvisited URL’s prediction score by the score of the node in our concept similarity context graph, while improving the retrieval precision and recall ratio. Firstly, a concept lattice is constructed using the visited pages, and then extract the core concepts which reflect the user’s query topic. Each of the unvisited URLs is assigned a prediction score which is used to determine which URL should be crawled firstly.

In the future works, we will study an efficient algorithm by constructing concept lattice incrementally in order to save the time consuming in reconstructing the concept lattice, and prove our model by experiments.

Acknowledgements. This work is supported by the Education Department Foundation of Sichuan Province (Grant No.2006A086), the Application Foundation of Sichuan Province (Grant No.2006J13-056), the Cultivating Foundation of Science and Technology of Xihua University (Grant No.R0622611), the cultivating foundation of the science and technology leader of Sichuan province.

References

1. Gulli, A., Signorini, A.: The indexable Web is More Than 11.5 BillionPages. In: Proceedings of the 14th International Conference on WWW (WWW 2005), pp. 902–903 (2005)
2. Chakrabarti, S., Berg, M., Dom, B.: Focused Crawling: a New Approach to Topicspecific Web Resource Discovery. *Comput. Networks* 31, 1623–1640 (1999)
3. Ching-Chi, H., Fan, W.: Topic-specific Crawling on the Web with the Measurements of the Relevancy Context Graph. *Information Systems* 31, 232–246 (2006)
4. Almpantidis, G., Kotropoulos, C., Pitas, I.: Combining Text and Link Analysis for Focused Crawling—An Application for Vertical Search Engines. *Information Systems* 32, 886–908 (2007)
5. Rungsawang, A., Angkawattanawit, N.: Learnable Topic-specific Web Crawler. *Journal of Network and Computer Applications* 28, 97–114 (2005)
6. Ganter, B., Wille, R.: *Formal Concept Analysis: Mathematical Foundations*. Springer, Berlin (1999)
7. CREDO Web Site, <http://credo.fub.it/>
8. Anna, F.: Ontology-based Concept Similarity in Formal Concept Analysis. *Information Sciences* 176, 2624–2641 (2006)
9. Li, Y., Bandar, Z.A., McLean, D.: An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources. *On Knowledge and Data Engineering* 15, 871–882 (2003)
10. Du, Y.J.: *Study and Implement on Intelligent Action of Search Engine*. Ph.D. dissertation, Southwest Jiaotong University (2005)

An IACO and HPSO Method for Spatial Clustering with Obstacles Constraints

Xueping Zhang¹, Jiayao Wang², Dexian Zhang¹, and Zhongshan Fan³

¹ School of Information Science and Engineering, Henan University of Technology, Zhengzhou 450052, China

² School of Surveying and Mapping, PLA Information Engineering University, Zhengzhou 450052, China

³ Henan Academy of Traffic Science and Technology, Zhengzhou 450052, China
zhang_xpcn@yahoo.com.cn

Abstract. In this paper, we propose an Improved Ant Colony Optimization (IACO) and Hybrid Particle Swarm Optimization (HPSO) method for Spatial Clustering with Obstacles Constraints (SCOC). In the process of doing so, we first use IACO to obtain the shortest obstructed distance, and then we develop a novel HPKSCOC based on HPSO and K-Medoids to cluster spatial data with obstacles. The experimental results demonstrate that the proposed method, performs better than Improved K-Medoids SCOC in terms of quantization error and has higher constringency speed than Genetic K-Medoids SCOC.

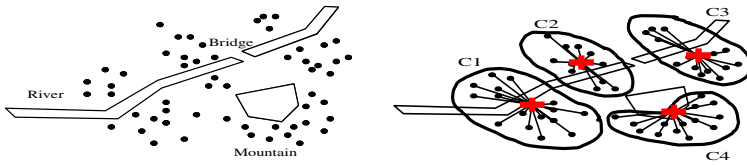
Keywords: Spatial Clustering; Obstacles Constraints, Ant Colony Optimization, Particle Swarm Optimization, K-Medoids Algorithm.

1 Introduction

Spatial Clustering with Obstacles Constraints (SCOC) has been a new topic in Spatial Data Mining (SDM). As an example, Fig.1 shows clustering spatial data with physical obstacle constraints. Ignoring the constraints leads to incorrect interpretation of the correlation among data points. To the best of our knowledge, only three clustering algorithms for SCOC have been proposed, that is COD-CLARANS [1], AUTOCLUST+ [2], and DBCluC [3,4], but many questions exist in them. Then we developed Genetic K-Medoids SCOC (GKSCOC) based on Genetic algorithms (GAs) and Improved K-Medoids SCOC (IKSCOC) in [5]. The experiments show that GKSCOC is effective but the drawback is a comparatively slower speed in clustering.

In this paper, we propose an Improved Ant Colony Optimization (IACO) and Hybrid Particle Swarm Optimization (HPSO) method for SCOC. The experimental results demonstrate that the proposed method performs better than IKSCOC in terms of quantization error and has higher constringency speed than GKSCOC.

The remainder of the paper is organized as follows. Section 2 discusses using IACO to get the optimal obstructed distance and HPKSCOC is presented in Section 3. The performances of HPKSCOC implementation on datasets are showed in Section 4 and Section 5 concludes the paper.



(a) Data objects and obstacles constraints (b) Clusters ignoring obstacle constraints

Fig. 1. Clustering data objects with obstacles constraints

2 Using IACO to Get the Optimal Obstructed Distance

2.1 Ant Colony Optimization

ACO has been inspired by the observation on real ant colony’s foraging behavior. The first ACO system was introduced by Marco Dorigo in 1992 [6]. ACO algorithm has two important features. One is the positive feedback process shown in the group behavior of ant colony. The other feature is the excellent distributed parallel computing ability. In many references, it has been proved that ACO algorithm has very preferable ability in finding the better solutions.

2.2 Optimal Obstructed Distance by Improved ACO

Definition 1. (Obstructed distance) Given point p and point q , the obstructed distance $d^o(p, q)$ is defined as the length of the shortest Euclidean path between two points p and q without cutting through any obstacles.

Let W be the sum of ants; n be the number of partition modules in two dimensions; $d_{ij}^o(i, j = 1, 2, \dots, n)$ be the obstructed distance between point i and point j ; $b_i(t)$ be the number of ants on point i at moment t ; $\tau_{i,j}(t)$ be the quantity of the residual pheromone path $\langle i, j \rangle$ at moment t . Then we have equation $w = \sum_{i=1}^n b_i(t)$. The transition probability which ant $k(k = 1, 2, \dots, W)$ wants to transit from point i to j at moment t can be defined as:

$$P_{i,j}^k(t) = \begin{cases} \frac{(\tau_{i,j}^\alpha(t) \cdot \eta_{i,j}^\beta(t))}{\left(\sum_{r \in S_i^k} (\tau_{i,r}^\alpha(t) \cdot \eta_{i,r}^\beta(t))\right)} & \text{if } j \in S_i^k \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $\eta_{i,j}(t)$ is a local heuristic function and it can be defined as $1/d_{ij}^o$ in our discussed problem; α and β determine the relative weightiness of the quantity of $\tau_{i,j}(t)$ and $\eta_{i,j}(t)$, respectively; S_i^k denotes the feasible neighboring region of ant k on point i . Here, α and β will be modified based on the time variation so as to obtain the better solutions [6]:

$$\alpha = \begin{cases} (4q)/m & 0 \leq q < m \\ 4 & m \leq q \leq u \end{cases} \tag{2}$$

$$\beta = \begin{cases} (3m - 1.5q)/m & 0 \leq q < m \\ 1.5 & m \leq q \leq u \end{cases} \tag{3}$$

where m is the critical moment and u is the pre-established terminative moment.

After h moments, the quantity of pheromone trail on every path will be modified based on the following equations:

$$\tau_{i,j}(t+h) = \rho \cdot \tau_{i,j}(t) + \sum_{k=1}^w \Delta \tau_{i,j}^k \tag{4}$$

where ρ denotes the degree of redundancy of pheromone trail after it volatilized on certain path; $\Delta \tau_{i,j}^k$ is the quantity of pheromone trail released by ant k on path $\langle i, j \rangle$ in this circle, and its definition is as the following:

$$\Delta \tau_{i,j}^k = \begin{cases} Q/L_k & \text{if ant } k \text{ passes the} \\ & \text{path } \langle i, j \rangle \text{ in this circle} \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where Q is a numerical constant that denotes the intensity of pheromone trail and L_k denotes the length of all paths visited by ant k in this circle. $\phi_{i,j}^k$ can be defined as:

$$\phi_{i,j}^k = \begin{cases} \frac{\left((Maxd_{A(i),e}^o - d_{j,e}^o) \cdot \omega + \mu \right)^\lambda}{\sum_{j \in Available(i)} \left((Maxd_{A(i),e}^o - Dist\ an\ ce_{j,e}) \cdot \omega + \mu \right)^\lambda} & \text{if } j \in Available(i) \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $Available(i)$ is the set of points that are around the point i , which are not in obstacle area and within one unit distance. Here, we define that the ant can only move in four directions: front, back, left, and right. Then we have inequation $0 < Available(i) \leq 4$. $d_{j,e}^o$ is the obstructed distance from point j to point e .

$Maxd_{A(i),e}^o$ is the maximum value of all $d_{j,e}^o$.

The IACO algorithm is adopted as follows. The simulation result is in Fig.2 and Fig.3.

1. Initialize feasible paths according to equation (6);
2. For $t = 1$ to t_{max} do {
3. For every path $\langle i, j \rangle$ do {
4. Calculate $length(\langle i, j \rangle)$;

5. Calculate $\Delta \tau_{i,j}^k, \tau_{i,j}(t+h)$ according to equation (5) and equation (4);
6. Straighten path $\langle i, j \rangle$ and Calculate $shortest(\langle i, j \rangle)$ and adjust $\tau_{i,j}(t+h)$ }
7. Generate feasible paths from point i to point j according to equation (6)
8. Output the shortest obstructed distance $d^o(i, j)$.

where t_{max} is the maximum number of iterations.

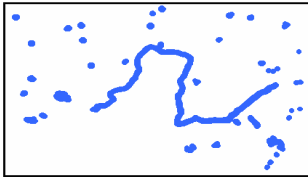


Fig. 2. Obstacle constraints

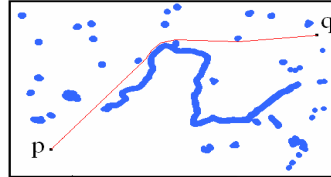


Fig. 3. Optimal obstructed path

3 Hybrid Algorithm Based PSO and GA

3.1 Standard PSO

PSO is a parallel population-based computation technique proposed by Kennedy and Eberhart in 1995. The mathematic description of PSO is as the following. Suppose the dimension of the searching space is D , the number of the particles is n . Vector $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ represents the position of the i^{th} particle and $pBest_i = (p_{i1}, p_{i2}, \dots, p_{iD})$ is its best position searched by now, and the whole particle swarm's best position is represented as $gBest = (g_1, g_2, \dots, g_D)$. Vector $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ is the position change rate of the i^{th} particle. Each particle updates its position according to the following formulas:

$$v_{id}(t+1) = wv_{id}(t) + c_1 rand() [p_{id}(t) - x_{id}(t)] + c_2 rand() [g_d(t) - x_{id}(t)] \tag{7}$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1), \quad 1 \leq i \leq n, \quad 1 \leq d \leq D \tag{8}$$

where c_1 and c_2 are positive constant parameters, $Rand()$ is a random function with the range $[0, 1]$, and w is the inertial function, in this paper, the inertial weight is set to the following equation.

$$w = w_{max} - \frac{w_{max} - w_{min}}{I_{max}} \times I \tag{9}$$

where w_{max} is the initial value of weighting coefficient, w_{min} is the final value of weighting coefficient, I_{max} is the maximum number of iterations or generation, and I is the current iteration or generation number. This process is repeated until

user-defined stopping criteria are satisfied. A disadvantage of the global PSO is that it tends to be trapped in a local optimum under some initialization conditions.

3.2 Hybrid PSO with GA Mutation

To make the algorithm have a good performance, we adopt a hybrid algorithm of PSO and GA with self-adaptive velocity mutation [7], named HPSO.

$$\frac{c_1 + c_2}{2} - 1 < w < 1 \quad \text{and} \quad c_1 + c_2 > 0 \tag{10}$$

Because w, c_1, c_2 have a constriction as equation (10), the following objective function is introduced to evaluate the particle performance of HPSO.

$$E(t) = - \sum_{k=1}^Q \frac{Z_k}{S} \ln \frac{Z_k}{S} \quad k = 1, 2, \dots, Q \tag{11}$$

where $E(t)$ is the particle population distribution entropy.

Here, the HPSO is adopted as follows.

1. Initialize swarm population, each particle's position and velocity;
2. Initialize $gBest, pBest, w_{max}, w_{min}, c_1, c_2$;
3. While (generation < maximum generation) do {
4. Generate next swarm by Equation (7) and Equation (8);
5. Update $gBest$ of swarm and $pBest$ of the particle;
6. For GA Initialize n_p, p_c, p_m , makespan T_G ;
7. Generate the initialization population.
8. While ($T < T_G$) do {
9. Calculate fitness of GA by Equation (11) and Selection candidate;
10. Crossover, Mutation, and Generate next generation}
11. Accept w, c_1, c_2 ;
12. if ($|v_{id}| > VMAX_d$) then $v_{id} = rand()VMAX_d \quad pBest(t) = x_{id}(t)$;
13. if $\|v\| \leq \epsilon$, terminate}
14. Output optimization results.

4 Spatial Clustering with Obstacles Constraints Based on HPSO and K-Medoids

4.1 IKSCOC Based on K-Medoids

Here, K-Medoids algorithm is adopted for SCOC to avoid cluster center falling on the obstacle. Square-error function is adopted to estimate the clustering quality.

$$E = \sum_{j=1}^{N_C} \sum_{p \in C_j} (d(p, m_j))^2 \tag{12}$$

where N_c is the number of cluster C_j , m_j is the cluster centre of cluster C_j , $d(p, q)$ is the direct Euclidean distance between the two points p and q . To handle obstacle constraints, accordingly, criterion function can be revised as:

$$E_o = \sum_{j=1}^{N_c} \sum_{p \in C_j} (d_o(p, m_j))^2 \tag{13}$$

where $d_o(p, q)$ is the obstructed distance between point p and point q .

The method of IKSCOC is adopted as follows [5].

1. Select N_c objects to be cluster centers at random;
2. Calculate E_o according to equation (13);
3. While (E_o changed) do {Let current $E = E_o$;
4. Select a not centering point to replace the cluster center m_j randomly;
5. Calculate E according to equation (12);
6. If $E >$ current E , go to 4
7. Calculate E_o ;
8. If $E_o <$ current E , form new cluster centers}.

4.2 HPKSCOC Based on HPSO and K-Medoids

PSO has been applied to data clustering [8]. In the context of clustering, each particle X_i is constructed as follows:

$$X_i = (m_{i1}, \dots, m_{ij}, \dots, m_{iN_c}) \tag{14}$$

where m_{ij} refers to the j^{th} cluster centroid of the i^{th} particle in cluster C_{ij} . And then the objective function is defined as follows:

$$f(x_i) = \frac{1}{\sum_{j=1}^{N_c} \sum_{p \in C_{ij}} d_o(p, m_j)} \tag{15}$$

The HPKSCOC is developed as follows.

1. Execute IKSCOC to initialize one particle to contain N_c cluster centroids;
2. Initialize the other particles to contain N_c cluster centroids at random;
3. For $t = 1$ to t_{max} do {
4. For each particle X_i do {
5. For each object p do {
6. Calculate $d_o(p, m_{ij})$;
7. Assign object p to cluster C_{ij} such that $d_o(p, m_{ij}) = \min_{c=1, \dots, N_c} \{d_o(p, m_{ic})\}$;

8. Calculate the fitness according to equation (15) ;}
9. Update $gBest$ and $pBest_i$;
10. For GA initialize n_p , p_c , p_m , T_G ;
11. Generate the initialization population.
12. While $T < T_G$ do {
- 13 Calculate fitness of GA by equation (11) and Select candidate;
14. Crossover, Mutation, and Generate next generation}
15. Accept w, c_1, c_2 and Update cluster centroids by equations (7) and (8);
16. if $\|v\| \leq \varepsilon$, terminate;
17. Optimize new individuals using IKSCOC ;}

where t_{\max} is the maximum number of iteration, ε is the minimum velocity. STEP 1 is to overcome the disadvantage of the global PSO which tends to be trapped in a local optimum under some initialization conditions. STEP 17 is to improve the local constringency speed of the global PSO.

5 Results and Discussion

We have made experiments separately by K-Medoids, IKSCOC, GKSCOC and HPKSCOC. $n = 50$, $w_{\max} = 0.999$, $w_{\min} = 0.001$, $c_1 = c_2 = 2$, $V_{\max} = 0.4$, $t_{\max} = 100$, $T_G = 0.01$, $n_p = 50$, $p_c = 0.6$, $p_m = 0.01$, $\varepsilon = 0.001$. Fig.4 shows the results on Dataset1. Fig.4 (a) shows the original data with simple obstacles. Fig.4 (b) shows the results of 4 clusters found by K-Medoids without considering obstacles constraints. Fig.4(c) shows 4 clusters found by IKSCOC. Fig.4 (d) shows 4 clusters found by GKSCOC. Fig.4 (e) shows 4 clusters found by HPKSCOC. Obviously, the results of the clustering illustrated in Fig.4(c), Fig.4 (d) and Fig.4 (e) have better practicalities than that in Fig.4 (b), and the ones in Fig.4 (e) and Fig.4 (d) are both superior to the one in Fig.4(c).

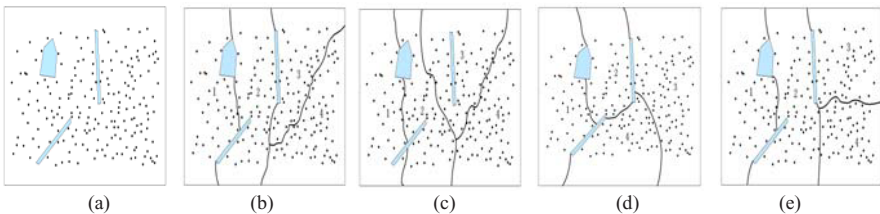


Fig. 4. Clustering Dataset1

Fig.5 is the value of J showed in every experiment on Dataset1 by IKSCOC and HPKSCOC respectively. It is showed that IKSCOC is sensitive to initial value and it constringes in different extremely local optimum points by starting at different initial value while HPKSCOC constringes nearly in the same optimum points at each time.

Fig.6 is the constringency speed in one experiment on Dataset1. It is showed that HPKSCOC constringes in about 12 generations while GKSCOC constringes in nearly 25 generations. So, it can be drawn that HPKSCOC is effective and has higher constringency speed than GKSCOC. Therefore, we can draw the conclusion that HPKSCOC has stronger global constringent ability than IKSCOC and has higher convergence speed than GKSCOC.

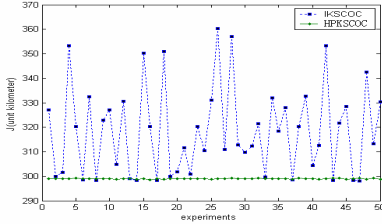


Fig. 5. HPKSCOC vs. IKSCOC

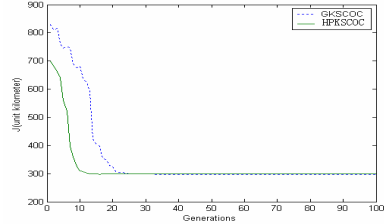


Fig. 6. HPKSCOC vs. GKSCOC

6 Conclusions

In this paper, we explore the applicability of ACO and HPSO for SCOC. The experimental results demonstrate the effectiveness and efficiency of the proposed method, which is better than IKSCOC in terms of quantization error and is superior to GKSCOC in speed.

Acknowledgments. This work is partially supported by the Science Technology Innovation Project of Henan (Number: 2008HASTIT012), the Natural Sciences Fund of Henan (Number: 0511011000, Number: 0624220081).

References

1. Tung, A.K.H., Hou, J., Han, J.: Spatial Clustering in the Presence of Obstacles. In: Proceedings of International Conference on Data Engineering (ICDE 2001), Heidelberg, Germany, pp. 359–367 (2001)
2. Estivill-Castro, V., Lee, I.J.: AUTOCLUST+: Automatic Clustering of Point-Data Sets in the Presence of Obstacles. In: Proceedings of the International Workshop on Temporal, Spatial and Spatial-Temporal Data Mining, Lyon, France, pp. 133–146 (2000)
3. Zaïane, O.R., Lee, C.H.: Clustering Spatial Data When Facing Physical Constraints. In: Proceedings of the IEEE International Conference on Data Mining (ICDM 2002), Maebashi City, Japan, pp. 737–740 (2002)
4. Wang, X., Rostoker, C., Hamilton, H.J.: DBRS+: Density-Based Spatial Clustering in the Presence of Obstacles and Facilitators (2004), <http://ftp.cs.uregina.ca/Research/Techreports/2004-09.pdf>
5. Zhang, X.P., Wang, J.Y., Wu, F., Fan, Z.S., Li, X.Q.: A Novel Spatial Clustering with Obstacles Constraints Based on Genetic Algorithms and K-Medoids. In: Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications (ISDA 2006), Jinan, Shangdong, China, pp. 605–610 (2006)

6. Maki, K., Habib, Asama, Hajime: Optimal Path Planning for Mobile Robots Based on Intensified Ant Colony Optimization Algorithm. In: Proceedings of the 2003 IEEE International Conference on Robotics, Intelligent Systems and Signal Processing, Changsha, China, pp. 131–136 (2003)
7. Esmín, A.A.A., Lambert-Torres, G., Alvarenga, G.B.: Hybrid Evolutionary Algorithm Based on PSO and GA Mutation. In: Proceedings of the 6th International Conference on Hybrid Intelligent Systems (HIS 2006), pp. 57–61 (2006)
8. Van der Merwe, D.W., Engelbrecht, A.P.: Data Clustering Using Particle Swarm Optimization. In: Proceedings of IEEE Congress on Evolutionary Computation 2003, pp. 215–220 (2003)

Attributes Reduct and Optimal Decision Rules Acquisition in Fuzzy Objective Information Systems

Fang Yang, Yan-Yong Guan, and Zhao-Xia Yu

School of Science, University of Jinan, Postfach 25 00 22,
106 Jiwei Road, Jinan, Shandong, P.R. China
ss_guanyy@ujn.edu.cn

Abstract. Definitions of three new types of attributes reducts of objects in fuzzy objective information systems are introduced respectively. The judgment theorems and discernibility functions with respect to these types of reducts are proposed, from which computing methods of attributes reduct of objects can be derived. Based on these reducts, three new types of optimal decision rules are obtained, and the degree of certainty of them are discussed. At last, three kinds of attributes reducts of the systems and their computing methods are given.

Keywords: Fuzzy information systems; Rough sets; Attribute reducts; Decision rules.

1 Introduction

Rough sets theory proposed by Pawlak [1] in 1982, is an effective mathematical tool for dealing with uncertain knowledge. It has achieved extensively applications in fields of artificial intelligence and knowledge acquisition.

Attributes reduct and decision rules acquisitions are main issues in the study of rough sets theory. Recently, there has been a lot of studying concerning the discrete information systems [2-4], but they are not suited to the information systems with fuzzy decisions.

For fuzzy objective information systems, Zhang, et. al [5] introduce the definitions of distribution reduct, maximum distribution reduct, approximation reduct, assignment reduct respectively, and propose the discernibility matrixes computing approaches respect to these reducts; Guan, et. al [6] present the α distribution reduct, α maximum distribution reduct, α assignment reduct, rough distribution reduct, and propose the discernibility matrixes computing approaches respect to these reducts; Huang, et. al [7] introduce variable precision rough set in fuzzy objective information systems and define β upper and lower approximation reduct, β upper and lower distribution reduct.

In this paper, we are concerned with the attributes reduct and optimal decision rules acquisition in fuzzy objective information systems. We first introduce basic notations, then propose definitions of three new types of attributes reducts respectively. We also give the judgment theorems and discernibility functions with respect to these types of reducts, from which attributes reduct can be computed. Based on these reducts, we present three new types of optimal decision rules, and degrees of certainty of them are discussed. At last, we discuss reduct of systems.

2 Basic Notations

Definition 1. A fuzzy objective information system is a quadruple $S = \{U, A, F, G\}$, where $U = \{x_1, x_2, \dots, x_n\}$ is a non-empty finite set of objects called the universe; $A = \{a_1, a_2, \dots, a_m\}$ is a nonempty finite set of attributes; $F = \{f_l : U \times A \rightarrow V_l, l \leq m\}$, where $f_l(x, a_l)$ is the value of a_l on $x \in U$, V_l is the domain of $a_l, a_l \in A$; We denote $f_l(x, a_l) = a_l(x)$ for convenience; $G = \{\tilde{d}_1, \tilde{d}_2, \dots, \tilde{d}_r\}$, where $\tilde{d}_j \in F(U) (j \leq r)$ and $\tilde{d}_j(x_k) \in [0, 1] (k \leq n, j \leq r)$, and $F(U)$ denote the set of all fuzzy sets on U .

When $r = 1$, $S = \{U, A, F, \{\tilde{d}\}\}$ is referred to as a fuzzy single-objective information system. We are only concerned with fuzzy single-objective information systems as presented in Table 1 in this paper.

For $B \subseteq A$, an indiscernibility relation on U is defined as follows:

$$R_B = \{(x, y) \in U \times U : a_l(x) = a_l(y), \forall a_l \in B\} . \tag{1}$$

R_B is an equivalent relation on U . For $\forall x \in U$, the equivalent class containing x is denoted by $[x]_B = \{y \in U \mid (x, y) \in R_B\}$, the partition of U by R_B is $U / R_B = \{[x]_B \mid x \in U\}$.

Definition 2. ([5]). Define

$$\underline{R}_B(\tilde{d})(x) = \min\{\tilde{d}(y) \mid y \in [x]_B\}, \quad \overline{R}_B(\tilde{d})(x) = \max\{\tilde{d}(y) \mid y \in [x]_B\} . \tag{2}$$

$\underline{R}_B(\tilde{d})$ and $\overline{R}_B(\tilde{d})$ are referred to as the lower and upper approximations of the fuzzy decision \tilde{d} with respect to attribute B respectively. $(\underline{R}_B(\tilde{d}), \overline{R}_B(\tilde{d}))$ is referred to as the rough fuzzy set of \tilde{d} with respect to B .

Table 1. A fuzzy single-objective information system

U	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
a_1	2	3	2	2	1	1	3	1	2	3
a_2	1	2	1	2	1	1	2	1	1	2
a_3	3	1	3	3	4	2	1	4	3	1
\tilde{d}	0.2	0.7	0.4	0.8	0.3	0.5	1.0	0.4	0.3	0.9

3 Reduct of Objects and Optimization of Decision Rules

Definition 3. Let $S = \{U, A, F, \{\tilde{d}\}\}$ be a fuzzy objective information system, for $B \subseteq A$ and $\forall x \in U$, x generates three kinds of fuzzy decision rules as follows:

$$\begin{aligned} r_x &: \bigwedge_{a \in B} (a, a(x)) \rightarrow (\tilde{d} \geq \underline{R}_B(\tilde{d})(x)) , \\ \overline{r}_x &: \bigwedge_{a \in B} (a, a(x)) \rightarrow (\tilde{d} \leq \overline{R}_B(\tilde{d})(x)) , \\ r_x &: \bigwedge_{a \in B} (a, a(x)) \rightarrow (\tilde{d} \in [\underline{R}_B(\tilde{d})(x), \overline{R}_B(\tilde{d})(x)]) , \end{aligned}$$

we call \underline{r}_x (\overline{r}_x, r_x) lower (upper, interval) approximation fuzzy decision rule.

Definition 4. In a fuzzy objective information system $S = \{U, A, F, \{\tilde{d}\}\}$, for $\forall x \in U$, we define:

$$\underline{D}(x) = |R_B(\tilde{d})(x) - 0|; \overline{D}(x) = |1 - \overline{R}_B(\tilde{d})(x)|; D(x) = |1 - \overline{R}_B(\tilde{d})(x) - R_B(\tilde{d})(x)|. \quad (3)$$

and call $\underline{D}(x)$ ($\overline{D}(x)$, $D(x)$) lower (upper, interval) approximation accurate degree of x respectively.

It can be shown that the more $\underline{D}(x)$ ($\overline{D}(x)$, $D(x)$) is, the higher degree of certainty of r_x (\overline{r}_x , r_x) is. Especially, when $\underline{D}(x) = 1$ ($\overline{D}(x) = 1$, $D(x) = 1$), we call r_x (\overline{r}_x , r_x) a definite lower (upper, interval) approximation fuzzy decision rule.

As of the optimization of decision rules, it often defined as a process of simplifying the description of condition attributes of rules under the decision conclusions keep unchanged, so that the necessary condition attributes for decision conclusions could be found.

Definition 5. In fuzzy objective information system $S = \{U, A, F, \{\tilde{d}\}\}$, for $B \subseteq A$ and $x \in U$, if $R_B(\tilde{d})(x) = R_A(\tilde{d})(x)$, we call B a lower approximation simplification of x ; furthermore, if $R_L(\tilde{d})(x) \neq R_B(\tilde{d})(x)$ for $\forall L \subset B$ ($L \neq B$), we call B a lower approximation reduct of x .

If $\overline{R}_B(\tilde{d})(x) = \overline{R}_A(\tilde{d})(x)$, we call B an upper approximation simplification of x ; furthermore, if $\overline{R}_L(\tilde{d})(x) \neq \overline{R}_B(\tilde{d})(x)$ for $\forall L \subset B$ ($L \neq B$), we call B an upper approximation reduct of x .

If $[R_B(\tilde{d})(x), \overline{R}_B(\tilde{d})(x)] = [R_A(\tilde{d})(x), \overline{R}_A(\tilde{d})(x)]$, we call B an interval approximation simplification of x ; furthermore, if $[R_L(\tilde{d})(x), \overline{R}_L(\tilde{d})(x)] \neq [R_B(\tilde{d})(x), \overline{R}_B(\tilde{d})(x)]$ for $\forall L \subset B$ ($L \neq B$), we call B an interval approximation reduct of x .

If B is a lower (upper, interval) approximation simplification of x , we call $\bigwedge_{a \in B} (a, a(x)) \rightarrow (\tilde{d} \geq \underline{R}_A(\tilde{d})(x))$ ($\bigwedge_{a \in B} (a, a(x)) \rightarrow (\tilde{d} \leq \overline{R}_A(\tilde{d})(x))$), $\bigwedge_{a \in B} (a, a(x)) \rightarrow (\tilde{d} \in [R_A(\tilde{d})(x), \overline{R}_A(\tilde{d})(x)])$) a simplified lower (upper, interval) approximation decision rule of $\bigwedge_{a \in A} (a, a(x)) \rightarrow (\tilde{d} \geq \underline{R}_A(\tilde{d})(x))$ ($\bigwedge_{a \in A} (a, a(x)) \rightarrow (\tilde{d} \leq \overline{R}_A(\tilde{d})(x))$), $\bigwedge_{a \in A} (a, a(x)) \rightarrow (\tilde{d} \in [R_A(\tilde{d})(x), \overline{R}_A(\tilde{d})(x)])$); if B is a lower (upper, interval) approximation reduct of x , we call $\bigwedge_{a \in B} (a, a(x)) \rightarrow (\tilde{d} \geq \underline{R}_A(\tilde{d})(x))$ ($\bigwedge_{a \in B} (a, a(x)) \rightarrow (\tilde{d} \leq \overline{R}_A(\tilde{d})(x))$), $\bigwedge_{a \in B} (a, a(x)) \rightarrow (\tilde{d} \in [R_A(\tilde{d})(x), \overline{R}_A(\tilde{d})(x)])$) an optimal lower (upper, interval) approximation decision rule.

By Definition 5, we can obtain following Proposition 1:

Proposition 1. The lower (upper, interval) approximation reduct of x defined above keep lower (upper, interval) approximation of decision class unchanged respectively.

For $\forall (x, y) \in U \times U$, denote

$$\alpha(x, y) = \{a_i \in A \mid f_i(x, a_i) \neq f_i(y, a_i)\}. \quad (4)$$

then $\alpha(x, y)$ is the set of attributes which can discern x and y . It is obvious that $\alpha(x, y) = \emptyset \Leftrightarrow [x]_A = [y]_A$. $M = (\alpha(x, y))_{(x,y) \in U \times U}$ is called discernibility matrix of S .

Corollary 1. For $B \subseteq A$ and $x \in U$, we can obtain

$$\underline{R}_B(\tilde{d})(x) \leq \underline{R}_A(\tilde{d})(x), \overline{R}_B(\tilde{d})(x) \geq \overline{R}_A(\tilde{d})(x).$$

Theorem 1. Let S be a fuzzy objective information system, for $B \subseteq A$ and $x, y \in U$,

1. $\underline{R}_B(\tilde{d})(x) = \underline{R}_A(\tilde{d})(x)$ if and only if

$$\underline{R}_A(\tilde{d})(y) < \underline{R}_A(\tilde{d})(x) \Rightarrow \alpha(x, y) \cap B \neq \emptyset;$$
2. $\overline{R}_B(\tilde{d})(x) = \overline{R}_A(\tilde{d})(x)$ if and only if

$$\overline{R}_A(\tilde{d})(y) > \overline{R}_A(\tilde{d})(x) \Rightarrow \alpha(x, y) \cap B \neq \emptyset;$$
3. $[\underline{R}_B(\tilde{d})(x), \overline{R}_B(\tilde{d})(x)] = [\underline{R}_A(\tilde{d})(x), \overline{R}_A(\tilde{d})(x)]$ if and only if

$$(\underline{R}_A(\tilde{d})(y) < \underline{R}_A(\tilde{d})(x) \text{ or } \overline{R}_A(\tilde{d})(y) > \overline{R}_A(\tilde{d})(x)) \Rightarrow \alpha(x, y) \cap B \neq \emptyset.$$

Proof. We only prove the case of (1), the proofs of remains are similar.

" \Rightarrow " Assume $\underline{R}_A(\tilde{d})(y) < \underline{R}_A(\tilde{d})(x)$ holds but $\alpha(x, y) \cap B = \emptyset$, then $[y]_B = [x]_B$, and we can obtain that $\underline{R}_B(\tilde{d})(y) = \underline{R}_B(\tilde{d})(x)$. Since $\underline{R}_B(\tilde{d})(x) = \underline{R}_A(\tilde{d})(x)$ holds by condition assumption, we have $\underline{R}_B(\tilde{d})(y) = \underline{R}_A(\tilde{d})(x)$. By corollary 1, $\underline{R}_B(\tilde{d})(y) \leq \underline{R}_A(\tilde{d})(y)$, hence we have $\underline{R}_A(\tilde{d})(y) \geq \underline{R}_A(\tilde{d})(x)$. This is contradictive to the assumption $\underline{R}_A(\tilde{d})(y) < \underline{R}_A(\tilde{d})(x)$.

Therefore, we proved that $\underline{R}_A(\tilde{d})(y) < \underline{R}_A(\tilde{d})(x) \Rightarrow \alpha(x, y) \cap B \neq \emptyset$.

" \Leftarrow " For $y \in [x]_B$, $[y]_B = [x]_B$ holds, hence $\alpha(x, y) \cap B = \emptyset$, so $\underline{R}_A(\tilde{d})(y) \geq \underline{R}_A(\tilde{d})(x)$ must hold by condition assumption. Therefore we have

$$\underline{R}_B(\tilde{d})(x) = \min_{y \in [x]_B} \tilde{d}(y) = \min_{[y]_B = [x]_B} \tilde{d}(y) \geq \min_{[y]_B = [x]_B} \underline{R}_A(\tilde{d})(y) \geq \min_{[y]_B = [x]_B} \underline{R}_A(\tilde{d})(x) = \underline{R}_A(\tilde{d})(x)$$

On the other hand, we have $\underline{R}_B(\tilde{d})(x) \leq \underline{R}_A(\tilde{d})(x)$ by corollary 1.

So, we can obtain $\underline{R}_B(\tilde{d})(x) = \underline{R}_A(\tilde{d})(x)$.

Definition 6. In a fuzzy objective information system S , we denote

$$\begin{aligned} \underline{\Delta}(x) &= \bigwedge_{R_A(d)(y) < R_A(d)(x)} (\vee \alpha(x, y)), \\ \overline{\Delta}(x) &= \bigwedge_{R_A(d)(y) > R_A(d)(x)} (\vee \alpha(x, y)), \\ \Delta(x) &= \underline{\Delta}(x) \wedge \overline{\Delta}(x). \end{aligned} \tag{5}$$

and call $\underline{\Delta}(x)(\overline{\Delta}(x), \Delta(x))$ lower (upper, interval) approximation discernibility function of x respectively. Especially, we define $\wedge\phi = \vee A = \vee_{a_i \in A} a_i$ for vacuous set ϕ .

By Boolean reasoning techniques, from Theorem 1 and Definition 6, we obtain following proposition2:

Proposition 2. B is a lower (upper, interval) approximation reduct of $x \Leftrightarrow \wedge B$ is a prime implicant of $\underline{\Delta}(x)(\overline{\Delta}(x), \Delta(x))$.

Example 1. We will illustrate approaches to compute three types of reducts and optimal decision rules for the fuzzy decision presented in Table 1.

We have $U/R_A = \{[x_1]_A, [x_2]_A, [x_4]_A, [x_5]_A, [x_6]_A\}$, where $[x_1]_A = \{x_1, x_3, x_9\}$, $[x_2]_A = \{x_2, x_7, x_{10}\}$, $[x_4]_A = \{x_4\}$, $[x_5]_A = \{x_5, x_8\}$, $[x_6]_A = \{x_6\}$.

We will take $[x_4]_A = \{x_4\}$ for example. By definition 6, we have:

$$\underline{\Delta}(x_4) = (a_1 \wedge a_2) \vee (a_2 \wedge a_3), \quad \overline{\Delta}(x_4) = a_1 \vee a_3, \quad \Delta(x_4) = (a_1 \wedge a_2) \vee (a_2 \wedge a_3).$$

By proposition 2, lower (upper, interval) approximation reducts of x_4 are $\{a_1, a_2\}$ and $\{a_2, a_3\}$ ($\{a_1\}$ and $\{a_3\}$, $\{a_1, a_2\}$ and $\{a_2, a_3\}$).

The optimization of $\underline{r}_{x_4}(\overline{r}_{x_4}, r_{x_4})$ are:

$$\begin{aligned} (a_1, 2) \wedge (a_2, 2) &\rightarrow (d \geq 0.8), \quad (a_2, 2) \wedge (a_3, 3) \rightarrow (d \geq 0.8). \\ ((a_1, 2) \rightarrow (d \leq 0.8), (a_3, 3) &\rightarrow (d \leq 0.8)); \\ (a_1, 2) \wedge (a_2, 2) &\rightarrow (d \in [0.8, 0.8]), \quad (a_2, 2) \wedge (a_3, 3) \rightarrow (d \in [0.8, 0.8]). \end{aligned}$$

By definition 4, the degree of certainty of $\underline{r}_{x_4}(\overline{r}_{x_4}, r_{x_4})$ is $0.8(0.2, 1)$, so r_{x_4} is a definite fuzzy decision rule.

4 Reduct of Systems

Definition 7. Let S be a fuzzy objective information system, if $\underline{R}_B(\tilde{d})(x) = \underline{R}_A(\tilde{d})(x)$ ($\overline{R}_B(\tilde{d})(x) = \overline{R}_A(\tilde{d})(x)$, $[\underline{R}_B(\tilde{d})(x), \overline{R}_B(\tilde{d})(x)] = [\underline{R}_A(\tilde{d})(x), \overline{R}_A(\tilde{d})(x)]$) holds for $\forall x \in U$, then we call B a lower (upper, interval) approximation consistent set of S ; if B is a lower (upper, interval) approximation consistent set of S , and no proper subset of B is a lower (upper, interval) approximation consistent set of S , we call B a lower (upper, interval) approximation reduct of S .

By Definition 6 and 7, we have following Proposition 3:

Proposition 3. Let S be a fuzzy objective information system. 1. An interval approximation consistent set must be a lower (upper) approximation consistent set; 2. For $B \subseteq A$, if B is lower and upper approximations consistent set, B must be an interval approximation consistent set.

By Definition 6 and Theorem 1, we have following Theorem 2:

Theorem 2. Let S be a fuzzy objective information system. 1. B is a lower approximation consistent set if and only if $\underline{R}_A(\tilde{d})(y) < \underline{R}_A(\tilde{d})(x) \Rightarrow [x]_B \cap [y]_B = \emptyset$ for $\forall x, y \in U$; 2. B is an upper approximation consistent set if and only if $\overline{R}_A(\tilde{d})(y) > \overline{R}_A(\tilde{d})(x) \Rightarrow [x]_B \cap [y]_B = \emptyset$ for $\forall x, y \in U$; 3. B is an interval approximation consistent set if and only if $\underline{R}_A(\tilde{d})(y) < \underline{R}_A(\tilde{d})(x)$ or $(\overline{R}_A(\tilde{d})(y) > \overline{R}_A(\tilde{d})(x)) \Rightarrow [x]_B \cap [y]_B = \emptyset$ for $\forall x, y \in U$.

Definition 8. In a fuzzy objective information system S , let

$$\underline{\Delta} = \bigwedge_{x \in U} \underline{\Delta}(x), \quad \overline{\Delta} = \bigwedge_{x \in U} \overline{\Delta}(x), \quad \Delta = \bigwedge_{x \in U} \Delta(x) \tag{6}$$

we call them lower, upper and interval approximations discernibility function of S respectively.

By Definition 7, 8 and Theorem 2, we can obtain following Proposition 4:

Proposition 4. B is a lower (upper, interval) approximation reduct of $S \Leftrightarrow \wedge B$ is a prime implicant of $\underline{\Delta}(\overline{\Delta}, \Delta)$.

Example 2. We will illustrate approaches to compute three types of reducts of S for the fuzzy decision presented in Table 1.

By Definition 8, we obtain $\underline{\Delta} = a_2 \wedge a_3, \overline{\Delta} = a_2 \wedge a_3, \Delta = a_2 \wedge a_3$. So we obtain lower (upper, interval) approximation reduct of S is $\{a_2, a_3\}$.

5 Conclusions

This paper has studied the attributes reduct and optimal decision rules acquisition in fuzzy objective information systems. Three types of attributes reducts of objects such as lower approximation reduct, upper approximation reduct and interval approximation reduct are introduced respectively. The judgment theorems and discernibility functions with respect to these types of reducts are obtained, from which attributes reduct of objects in fuzzy objective information systems can be computed. Based on these reducts, three types of optimal decision rules are obtained; knowledge hidden in fuzzy objective information systems may be unraveled and can be represented by decision rules. In further research, we will develop the proposed approaches to more generalize and more complicated information systems such as the information systems with both the condition and decision attribute values being fuzzy.

Acknowledgements. This research is supported by the Scientific Research and Development Project of Shandong Provincial Education Department, China (J06P01), and Doctoral Foundation of University of Jinan, China (B0616).

References

1. Pawlak, Z.: Rough Sets. *International Journal of Computer and Information Science* 11, 341–356 (1982)
2. Kryszkiewicz, M.: Comparative Study of Alternative Type of Knowledge Reduct in Inconsistent Systems. *International Journal of Intelligent Systems* 16, 105–120 (2001)
3. Guan, Y.Y., Shi, K.Q., Xue, P.J.: Attributes Reduct and Decision Rule Optimization Based on Descriptors in Information Systems. *Chinese Journal of Control and Decision* 21(7), 787–791 (2006)
4. Guan, Y.Y., Li, J.P., Wang, Y.: Improved Discernibility Matrices on Decision Table. *International Journal of Advances in Systems Science and Applications* 6(3), 439–445 (2006)
5. Zhang, M., Zhang, W.X.: Knowledge Reducts in Information Systems with Fuzzy Decisions. *Chinese Journal of Engineering Mathematics* 20(2), 53–58 (2003)
6. Guan, T., Feng, B.Q.: Knowledge Reducts Methods in Fuzzy Objective Information Systems. *Chinese Journal of Software* 15(10), 1470–1478 (2004)
7. Huang, B., Zhou, X.Z., Hu, Z.J.: VPRS Model and Its Knowledge Reducts in Fuzzy Objective Information Systems. *Chinese Journal of Systems Engineering and Electronics* 29(11), 1859–1862 (2007)

Cross Sentence Alignment for Structurally Dissimilar Corpus Based on Singular Value Decomposition

Anna Ho, Fai Wong, Francisco Oliveira, and Yiping Li

Faculty of Science and Technology, University of Macau, Macau
{ma36560, derekfw, olifran, ypli}@umac.mo

Abstract. Extracting the alignment pairs is a critical step for constructing bilingual corpus knowledge base for Example Based Machine Translation Systems. Different methods have been proposed in aligning parallel corpus between two different languages. However, most of them focus on structurally similar languages like English-French. This paper presents a method of cross aligning Portuguese-Chinese bilingual comparable corpus. The proposed approach is based on Singular Value Decomposition techniques and similarity measurement, which covers the problem in aligning structurally dissimilar corpus and enhances the accuracy of the alignment result.

1 Introduction

Text alignment is a pre-process mainly used to build bilingual lexical resources which improve the quality of Machine Translation (MT). Basically, text alignment can be divided into four main different levels: paragraph, sentence, phrase and word [1]. Paragraph level and sentence level are procedures responsible in pairing groups of sentences and independent sentence respectively. Among different alignment levels, Chuang et al. [2] stated by aligning corpus at sentence level, this provides linguistic information that helps to build up a knowledge base to assist in natural language processing, such as multilingual information retrieval, phrase and word alignment analysis, collocation extraction, etc.

The types of bilingual corpus can be roughly divided into three types: parallel corpus, comparable corpus and quasi-parallel corpus [3]. When the bilingual documents are sentence-aligned and are exact translations to each other is known as parallel corpus, such as law statements and legal articles. If the bilingual documents are non-aligned sentences and bilingual content is nearly the exact translation to each other is known as comparable corpus, such as news or magazine articles. Quasi-parallel corpus is bilingual documents contain non-aligned sentence, non-translated to each other, with the same or different topic. With different types of input sources and alignment level, the applied techniques of the methods also need to be varied.

In this paper, we mainly focus on the solution of sentence alignment in Portuguese-Chinese which are structurally dissimilar language comparable corpus. In section 2, we will give a brief review of the related researches on alignment topics. Section 3 describes the proposed alignment algorithm based on Singular Value Decomposition technique. Section 4 describes the construction of Statistical Relationship Matrix. Then, the experiment results will be given in section 5. Finally, we conclude our research and discuss about the future improvements in section 6.

2 Related Works

Recently, existing literature already provides many different approaches on aligning sentences in bilingual documents. Those techniques can be generally classified into three main categories: lexical based, statistical based and combination of lexical and statistical based.

Lexical based method mainly makes use of lexical information, such as bilingual dictionary, to perform the alignment procedure. The research taken by Kay and Roscheisen [4] employs a partial alignment of the word level to introduce a maximum likelihood alignment of the sentence level. Chen's model [5] constructs a word-to-word translation model which estimates the cost of an alignment by dividing the bitext into sequence of sentence beads.

In statistical approach, Brown et al. [6] and Gale and Church [7] show a method of aligning sentences based on a statistical model of character length. Brown et al. make use of the major and minor anchor points to facilitate the alignment process. On the other hand, Gale and Church's model assigns a probabilistic score to each proposed correspondence of sentence base on the difference of their lengths and the variance of the difference.

Although statistical approach gives a high performance, it may not be suitable in cognate different languages, while lexical approach provides a lot of confirmation details of alignments but cannot solve the case with rare words or too large lexical information will cause low efficiency. Therefore, Li et al. [8] combine both statistical (length-based) and lexical methods in sentence alignment. Some lexical cues or parameters are included in order to maximum the result of a probability function.

However, the techniques mentioned in above cannot fulfill our cases. Unlike bitext from similar language families, Portuguese and Chinese are different in syntactic, appearance and grammars. The position of noun, adjective, adverb, etc are different from Portuguese and Chinese, and it is also not possible to do similarity measurement on words to assist in the alignment process as English-French case performs. Moreover, most of the literatures focus on parallel corpus which does not provide enough solutions for crossing alignment problem. With these concerns, a new approach should be investigated to deal with the new problems effectively.

3 Alignment Based on Single Value Decomposition (SVD)

In order to develop a method to improve the quality of cross alignment results in structurally dissimilar languages, we have to determine the main problems to face. In this paper, several principal problems have been studied and concluded.

3.1 Syntax Deviation

Cross alignment problem refers to situations when the sentence in any position of one language may not align to the same position of the sentence in target language. In real situation, MT system needs to handle the paraphrasis cases. To improve the system quality, knowledge base constructed with news or magazines articles can provide a better reference. Chuang et al. [2] states that the success in statistical approach

depends on the similarity of the bilingual languages family, and they are about 90% 1-1 match type for the aligned pairs. Therefore, in our case, we introduce the Singular Value Decomposition (SVD) technique to overcome the cross sentence alignment problem.

In a mathematical point of view, SVD is a technique to decompose a matrix W into the product of three matrices as shown in formula (1).

$$W = U \times S \times V^T \quad (1)$$

The columns of U and V are the left and right singular vectors describe the row entities and column entities of W respectively. S is a diagonal matrix contains singular values s_1, s_2, \dots, s_n which ordered in a descending way. There is also mathematical proof demonstrate that every matrix has its own decomposition. SVD has been applied into many areas such as latent semantic analysis (LSA) [9], sense disambiguation analysis [10], image processing and compression and gene expression analysis, etc. In these example application areas, they involve in filtering of useless information and relationship analysis. For example, LSA used to analyze the relationships between a set of documents and terms. It forms a term-document matrix which the entries are the occurrences of the terms in documents. SVD technique highlights the truly related terms and documents and even reveal the relationship between a term and its synonymies which does not show any relationship in the original matrix.

In our observation, we discover that there is a certain similarity between the crossing sentence alignment problem and LSA. Instead of finding relationships among the source terms and target terms, we need to find the relationships among Portuguese sentences and Chinese sentences. In LSA, the best related target term of a source term does not relate to any positional information, similarly, we need to match the sentence pairs with a crossing ability. If SVD assists LSA to filter out noisy information and improve the accuracy of the terms similarity measurement, this should also give a significant effect on our case.

3.2 Lexicalization

The next problem is about the deviation of Portuguese and Chinese linguistic information. We need to preserve the sequence of Chinese characters of a translation. This is because it is very often that a Chinese translation of a Portuguese word contains more than one Chinese character. For example “Abandonar” (leave) means “離開”, if the Chinese characters are not in the same sequence, such as “...離..開...” or “..開..離..”, they will have different meanings. If we count the above example as a correct translation for “Abandonar”, this will lower the accuracy of the aligner. Some meanings expressed in different Chinese words may actually refer to the same Portuguese word. For example, the word “correio” (post) can find a translation “郵包” in the dictionary, but in the corresponding bilingual sentence, it consists of the word “郵件” which cannot be found exactly in dictionary, though the expression in Chinese is a little bit different, but the meaning should be the same.

This problem can be solved in the lexical matching feature. The main concern of this feature consists of Full Match and Partial Match cases. By giving a basic Portuguese-Chinese dictionary, when a word in $Sentence_p$ can find a corresponding meaning from the dictionary and all the Chinese characters with the exact order of the

meaning exists in the target $Sentence_c$ is treated as a Full Match case. For those do not fulfill the Full Match case, we treat it as Partial Match case. To conclude these two cases, we define the following formula for this feature:

$$F_1(c, p) = \sum_k \arg \max_t \left(\frac{\sum_{i=1}^4 \frac{matched}{total} \times w_i}{\sum_{i=1}^4 nw_i} \right), w_i = gram_weight \quad (2)$$

The process calculates the lexical matching score for all the found meanings by sum up the highest score of the Chinese meaning set of a given Portuguese lexicon based on a modified 4-gram Chinese character approach. As shown in Eck et al. [11], N-gram has a characteristic of preserving sequential order which is suitable in our case. We have also encountered the sub-phrase problem in the estimation of basic relationship of sentence pairs. The case is defined as follow:

P1: Vamos encontrar-nos a porta do teatro/ C1: 我們在劇院門口
(We are at the cinema entrance)

P2: Vamos encontrar-nos a porta do teatro ou vou buscar-te a casa/ C2:
我們在劇院門口碰面還是我去你家接你

(Whether we meet at the cinema entrance or I go to join you at your house)

Ideally, P1 should align with C1 and P2 should align with C2. However, P1 is actually a sub-phrase of P2 and so as C1 and C2. Without considering this problem, the procedure will estimate the relationship between this two pairs wrongly. To correct the estimation, we introduce the penalty by applying the formula (3):

$$1 - \frac{non_matched_chinese_char}{total_chinese_char} \quad (3)$$

3.3 Deviation in Digit Representation

There are two sub-problems we need to face in digit problem. The first one is about the presentation of numbers or digits in Portuguese and Chinese and the second one is the preservation of the ordering of numbers in the group. When the groups of digits are not present in roman numbers in both languages, we cannot directly align the digits. Moreover, the units of digit presents differently between Portuguese and Chinese. For example, “億” cannot directly translate to “mil” or “milhão”, and the representation of unit is totally different. Though this kind of case may not exist in all types of corpus, once it exists, it is an important key to align the correct sentence pair. In our solution, we ignore the unit because we are only interested in the groups of digits and the decimal points. With formula (4), the probability statistics of digit groups among sentence pair can be calculated with guarantee that the digit groups are preserved as a whole.

$$\frac{\sum_{D \in \{Digit\}} Count(DigitGroup_C \in DigitGroup_P)}{Count(TotalDigitGroup)} \tag{4}$$

4 Construction of Statistical Relationship Matrix (SRM)

The architecture of SRM is based on idea of occurrences matrix in LSA. It is a sentence-by-sentence matrix. In our case the rows are the relationship vectors of each sentence, therefore we need to concatenate the Portuguese and Chinese sentence into one partition. There are four zones in SRM, Zone A consist of entries about only Portuguese sentence and Zone D for only Chinese sentences. These two zones have anchor characteristics for each relationship vectors. In fact, entries with identical sentence pair will assign one and others to be zero. For Zone B and Zone C, these are the entries with estimated relationship based on the feature scores of each sentence pair. The estimation is calculated by formula (5), where F_1 and F_4 are the lexical matching feature and digit group feature discussed in section 3 respectively, F_2 and F_3 are punctuation feature and length ratio measuring defined in formula (6) and formula (7) respectively.

$$W_1F_1 + W_2F_2 + W_3F_3 + W_4F_4 \leq 1 \tag{5}$$

$$\sum_j PuncMatch(punc_j, sentence_c) \leq 1 \tag{6}$$

$$1 - \begin{cases} 1, r \geq 1 \\ r \end{cases} \leq 1, r = \left| \frac{length_c}{length_p} - mean \right| \tag{7}$$

By applying SVD technique, SRM will be decomposed into three component matrices U, S and V . Dimensionality reduction of SRM, which is SRM' , can be approximated with the least and necessary rank. We choose the rank R as $\left\lceil \frac{k}{2} \right\rceil$ where k is the sum of sentence of Portuguese and Chinese, this rank is in fact representing approximately the number of sentences in the bi-text of one language so that we can ensure the result can cover the whole text. With this rank, S' only keeps R highest value of S , U' and V' only keeps R dimension instead of the original k dimension. As we have states that our process only interested on the similarity measurement of the rows, therefore the dimensionality reduction of SRM can just multiply U' and S' .

Based on the characteristics of SVD, it helps to find the relationship information among the sentence which is lack in provide from lexical matching and length ratio measurement of the sentence pair. Moreover, this also reduces the complex work in analyzing the grammatical problems between two languages through the alignment

process. By employing this technique should help to weaken the redundant relationship of a wrong sentence pair and strengthen the one of the correct, so that this can help to enhance the accuracy in cross sentence alignment.

Finally apply cosine similarity measurement to SRM' to construct the Alignment Matrix (AM), which is used for producing final alignment results. Table 1 illustrates the effective results of applying SVD by showing the cosine similarity measurement (before and after SVD) of one Chinese sentence between other Portuguese sentences.

Table 1. Effects of sentence pairs before and after applying SVD

	C1 : P1	C1 : P2	C1 : P3	C1 : P4
Before SVD	0.317767	0.175213	0.152894	0.164289
After SVD	0.968804	0.173242	0.096306	0.077471

From the above table, C1 should align with P1, therefore, the closeness value between C1 and P1 increase sharply after SVD. On the other hand, the closeness value between the other non-related sentences decreases. In order to ensure the result to be more reliable and accurate, the process filters out the similarity degree which less than a certain threshold, so that the remaining alignment pairs are said to be confirmed alignment.

5 Experiment Results

With this proposed framework, the experiments are carried out by using the following corpora: Macao Year Book 2006 (A), parts of Educational Digest Volume 14 and 17 (B), administrative laws of Macao SAR (C), Macao SAR Policy Address 2007 (D) and internal staff regulations of a company (E). All of the sources are provided with Portuguese and Chinese, covering magazines and reports with many crossing sentences, laws and regulations with less crossing sentences, from parallel to comparable types. In this way, we can evaluate the effectiveness in handling crossing problems. The following table shows the differences between with and without the application of SVD technique.

Table 2. Size of experimented corpus and improvements after applying SVD

Characteristics	Portuguese Sentences (Words)	Chinese Sentences (characters)	Sentences Captured	Accuracy (%)	
				Without SVD	With SVD
A Magazine	89 (2564)	72 (2945)	63	60.53	96.83
B Magazine	140 (3823)	108 (5024)	98	50.30	89.80
C Law	484 (7850)	484 (12148)	339	52.15	85.84
D Report	145 (3774)	107 (4707)	102	55.43	84.31
E Regulation	146 (3765)	132 (4888)	104	63.74	96.15

Threshold is a value that guarantees the method can capture 70% sentences of the corresponding corpus. Sentence pair with a similarity degree higher or equal to the threshold will be captured and is confirmed to be aligned pairs.

From the experiment results tables, they show the different accuracy gained between with and without applying SVD technique. The accuracy increases sharply with SVD technique. The main reason is that SVD filters out the noisy and useless relationship information among sentence pairs. Moreover, with the dimensionality reduction advantage, it makes the similar entries in the statistical relationship vector become more similar to each other and the dissimilar ones become more dissimilar.

6 Conclusions

In this paper, we have given a brief review of recent research based on lexical and statistical approach in similar and dissimilar language structure scenario. Moreover, we have stated the limitations of those researches in cross sentence alignment case. Therefore, by introducing the application of SVD, we have primarily showed that it can enhance the accuracy of cross level sentence alignment. However, the size of the initial matrix is quite big which affect the efficiency of the program when the corpus is big. We may design another data representation model which can keep the matrix in a smaller size. On the other hand, we can consider adding a Named Entity Recognition module, so that the method will not align wrongly for those sentences pairs with the same or similar meanings but with different names or places.

Acknowledgements. The research work reported in this paper was partially supported by “Fundo para o Desenvolvimento das Ciências e da Tecnologia” (Science and Development Fund) under grant 041/2005/A and Center of Scientific and Technological Research (University of Macau) under Cativo: 5571.

References

1. Manning, C., Schütze, H.: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge (1999)
2. Chuang, T.C., You, G.N., Chang, J.S.: Adaptive Bilingual Sentence Alignment. In: Proceedings of the 5th Conference of the Association for Machine Translation in the Americas on Machine Translation: From Research to Real Users, pp. 21–30. Springer, London (2002)
3. Fung, P., Cheung, P.: Multi-level Bootstrapping for Extracting Parallel Sentences from a Quasi-Comparable Corpus. In: Proceedings of the 20th International Conference on Computational Linguistics, Switzerland, pp. 1051–1057 (2004)
4. Kay, M., Roscheisen, M.: Text-translation Alignment. In: Computational Linguistics, vol. 19, pp. 121–142. MIT Press, Cambridge (1993)
5. Chen, S.F.: Aligning Sentences in Bilingual Corpora Using Lexical Information. In: Proceedings of 31st Annual Meeting of the Association for Computational Linguistics, Columbus, pp. 9–16 (1993)

6. Brown, P.F., Lai, J.C., Mercer, R.L.: Aligning Sentences in Parallel Corpora. In: 29th Annual Meeting of the Association for Computational Linguistics, Berkeley, CA, USA, pp.169–176 (1991)
7. Gale, W.A., Church, K.W.: A Program for Aligning Sentences in Bilingual Corpora. In: Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics, Berkeley, CA, USA, pp.177–184 (1991)
8. Li, Y.P., Pun, C.M., Wu, F.: Portuguese-Chinese Machine Translation in Macao. In: Proceedings of Machine Translation Summit VII, Singapore, pp. 236–243 (1999)
9. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by Latent Ssemantic Analysis. *Journal of the American Society for Information Science* 41(6), 391–407 (1990)
10. Oliveira, F., Wong, F., Ho, A., Li, Y.P., Dong, M.C.: Overcoming Data Sparseness Problem in Statistical Corpus Based Sense Disambiguation. In: Proceedings of the 10th International Conference on Enhancement and Promotion of Computational Methods in Engineering and Science, Sanya, Hainan, China, pp. 1099–1104 (2006)
11. Eck, M., Vogel, S., Waibel, A.: Low Cost Portability for Statistical Machine Translation Based on N-gram Frequency and TF-IDF. In: International Workshop on Spoken Language Translation 2005, Pittsburgh, USA (2005)

Event Detection and Tracking Based on Improved Incremental K-Means and Transductive SVM

Zhen Lei¹, Jianfeng Liao², Dong Li¹, and Lingda Wu³

¹ Department of Information Engineering,
Academy of Armored Force Engineering, Beijing 100072, China

² Department of Information Management,
Academy of Henan Economic Trade Vocation, Zhengzhou 450053, China

³ College of Information Systems and Management,

National University of Defense Technology, Changsha 410073, China

williamjohnmail@126.com, fengjianliao@tom.com, dongli62@tom.com,
wld@nudt.edu.cn

Abstract. This paper describes a new method for detecting events. The method utilizes density function to initialize cluster centers so as to get center points objectively, it can be used in both on-line detection and retrospective detection, and the quantity of clusters is affected little by the order in which the news stories are processed. This paper deals with event tracking with transductive support vector machine (TSVM). TSVM takes into account a particular test set and tries to minimize misclassifications of just those particular examples. The problem of effective density radius selection is discussed, and the performance differences between the event detection and tracking methods proposed in this paper and other methods are compared. The experimental results indicate that the methods proposed by this paper are effective and advanced.

Keywords: event detection, event tracking, density function, IIKM, TSVM.

1 Introduction

According to the latest statistic report of CNNIC, 84.38% of information got from Internet by Chinese users is news. Therefore, research on Internet news is becoming a hotspot in natural language processing (NLP) field. As a new direction of research on NLP, the aim of TDT[1] (Topic Detection and Tracking) study is to explore the modern way of identifying and following new or previously unidentified events in several continuous news streams.

The event detection is the problem of identifying stories in several continuous news streams that pertain to new or previously unidentified events. It is sub-divided into two forms. One is retrospective detection, which entails the discovery of previously unidentified events in an accumulated collection. The other is on-line detection, where the goal is to flag the onset of new events from live news feeds. For the purpose of our goal that finds out what happens in the past, the retrospective detection is the main direction to pursue, but flagging the onset of new events is also needed in some cases. This paper adopts IIKM[2] to detect events.

The event tracking is defined to be the task of associating incoming stories with events known to the system. A tracking system's goal is to correctly classify all of the subsequent stories. Automated tracking of events from chronologically ordered document streams is a new challenge for statistical text classification. Existing learning techniques must be adapted or improved in order to effectively handle difficult situations where the number of positive training instances per event is extremely smaller than negative instances and the majority of training documents are unlabelled. K nearest neighbor (KNN), Decision Trees and Rocchio are well-known methods suited to tracking task. However SVM is seldom used to tracking task. The main reason, we believe, is that the positive examples are extremely sparse in the training set and SVM is not always fit for learning task of small samples. In addition, event detection and tracking requires tracking systems offer a confidence measure, but a standard SVM produces the values that are not probabilities.

In order to overcome the limitation of SVM and use it in our tracking implementation, this paper employs TSVM[4] to finish tracking task. TSVM has substantially improved the already excellent performance of SVM, especially for small training sets. Finally we map the TSVM outputs into probabilities.

The structure of the paper is as follows. Section 2 and Section 3 detail the proposed event detection and tracking method used in our system respectively. Section 4 presents experimental results. Section 5 gives conclusions.

2 Detection Method Based on IIKM

The goal of an event detection system is to group together stories that discuss the same event and spotting new events as they happen.

2.1 Document Representation

Our approach uses conventional vector space model to represent the documents. Each document is represented by a vector of weighted terms, which can be either words or phrases.

Text subjects, especially subject concepts, precisely and concisely express the main topic and central idea of a whole document. Under this consideration, we give higher weight for subject strings. Normally, concepts in title are related to the news subject and more than 98% of titles can express the news subject clearly. We may obtain important subject strings such as who, what, where by finding the reappearance of some elements appearing in titles and bodies at the same time.

To compare the importance of different subject strings in the news, we need to get the weights of subject strings. Firstly, higher weights should be assigned to longer string. It is because that longer string includes more concrete information than shorter one. Secondly, higher weights should be assigned to high-frequency string. Normally most of news reporters often use some repeated concepts when they want to explain their important topic. Suppose there are k subject strings $TitSubStr_i$, $i = 1, 2, \dots, k$.

$TitSubStr_i$'s weight is normalized by the following formula:

$$w(TitSubStr_i)' = 3 * \frac{Len(TitSubStr_i) * Freq(TitSubStr_i)}{\max_{i \text{ in } k} \{Len(TitSubStr_i) * Freq(TitSubStr_i)\}} \tag{1}$$

Where $Len(TitSubStr)$ is the length of the subject string $TitSubStr$, and $Freq(TitSubStr_i)$ is the frequency of the subject string $TitSubStr$ appearing in the body. For the weight of term in body of d , we use traditional “l_{tc}” version of the $tf \cdot idf$ scheme.

2.2 Event Detection Method

There are lots of drawbacks to traditional incremental K-means in event detection, in order to overcome its shortcomings, this paper proposed an improved incremental K-means (IKM) for detecting events. The algorithm utilizes density function to initialize cluster centers to select initial cluster centers objectively, it can be used in both on-line detection and retrospective detection, and the quantity of clusters is affected little by the order in which the news stories are processed. The problem of effective density radius selection is discussed, and the performance difference between this method and other methods including Single-pass and traditional K-means.

The selection of initial cluster centers affects cluster results and convergence time greatly, unsuitable initial values often make results converge to an undesired minimum and affect convergence speed. In this paper, we adopt a density function[3] to initialize cluster centers. Firstly, the density function of sample x_i is defined as follows:

$$D_i^{(0)} = \sum_{k=1}^n \frac{1}{1 + f_d \|x_i - x_k\|^2} \tag{2}$$

Where

$$f_d = 4 / r_d^2 \tag{3}$$

Where r_d is valid density radius, it can be adopted in two forms:

$$r_f = \frac{1}{2} \min \{ \max \{ \|x_i - x_k\|, i = 1, \dots, n \}, k = 1, \dots, n \} \tag{4}$$

or

$$r_m = \frac{1}{2} \sqrt{\frac{1}{n(n-1)} \sum_{k=1}^n \sum_{i=1}^n \|x_i - x_k\|^2} \tag{5}$$

Let $D_1^* = \max \{ D_i^{(0)}, i = 1, \dots, n \}$, corresponding x_1^* is treated as the first initial cluster center, and the density functions of latter initial cluster centers is as follows:

$$D_i^{(k)} = D_i^{(k-1)} - D_k^* \frac{1}{1 + f_d \|x_i - x_k^*\|^2}, k = 1, \dots, c - 1 \quad (6)$$

where

$$D_k^* = \max\{D_i^{(k-1)}, i = 1, \dots, n\} \quad (7)$$

With the considerations described above, if inputted sample set $Z = \{x_1, x_2, \dots, x_m\}$, $x_i \in R^n$, $Flag_{final} = 0$, the IIKM-based event detection method used in our system is depicted as follows:

Step 1: for retrospective detection, anterior $m/5$ samples is joined to sample set Z , go to step 3.

Step 2: for retrospective detection, latter $m/8$ samples is joined to sample set Z , go to step 3; if residual unsettled sample number $N_{final} \leq m/8$, then $Flag_{final} = 1$ and these N_{final} samples are joined to sample set Z , yet go to step 3.

Step 3: density function is utilized to get K initial cluster centers $\{O_1, O_2, \dots, O_K\}$ of sample set Z , $ClusterNum \leftarrow K$.

Step 4: For each sample $x_i \in Z$, comparing the distance between each story and obtained cluster center to find cluster center O_i nearest to the story.

$$O_i \leftarrow \arg \min_i^{ClusterNum} Dist(x_i, O_i) \quad (8)$$

Step 5: if $Dist(x_i, O_i) < T_{Distance}$, x_i is joined to existed cluster $CCluster_i$.

$$CCluster_i \leftarrow CCluster_i \cup \{x_i\} \quad (9)$$

The center of $CCluster_i$ is also adjusted and go to step 7.

$$O_i \leftarrow \frac{n_i \times O_i + x_i}{n_i + 1}, n_i \leftarrow n_i + 1 \quad (10)$$

Otherwise, go to step 6.

Step 6: A new cluster is added.

$$ClusterNum \leftarrow ClusterNum + 1 \quad (11)$$

$$CCluster_{ClusterNum} \leftarrow \{x_i\} \quad (12)$$

$$O_{ClusterNum} \leftarrow x_i \quad (13)$$

Step 7:

$$Z \leftarrow Z - \{x_i\} \tag{14}$$

Step 8: If the cluster center is not steady yet, go to step 4, otherwise if $Flag_{final} = 0$, go to step 2 and if $Flag_{final} = 1$, the detection process is finished.

3 Tracking Method Based on TSVM

A tracking system's goal is to correctly classify all of the subsequent stories when they arrive. The principle of TSVM[4] is described as follows:

For a learning task, $P(\bar{x}, y) = P(y | \bar{x})P(\bar{x})$, the learner L is given a hypothesis space H of functions $h : X \rightarrow \{-1,1\}$ and an i.i.d. sample S_{train} of n training examples $(\bar{x}_1, y_1), \dots, (\bar{x}_n, y_n)$, $x \in X, y \in \{-1,+1\}$. In contrast to the inductive setting, the learner is also given an i.i.d. sample S_{test} of k test examples from the same distribution. The transductive learner L aims to select a function $h_L = L(S_{train}, S_{test})$ from H using S_{train} and S_{test} so that the expected number of erroneous predictions on the test examples is minimized.

$$R(L) = \int \frac{1}{k} \sum_{i=1}^k \Theta(h_L(\bar{x}_i^*, y_i^*)) dP(\bar{x}_1, y_1) \cdots dP(\bar{x}_k^*, y_k^*) \tag{15}$$

$\Theta(a, b)$ is zero if $a = b$, otherwise it is one.

For linearly non-separable case, the training process of TSVM can be described by the following optimization problem:

$$\left. \begin{aligned} & \text{Minimize over} \\ & (y_1^*, \dots, y_n^*, \bar{w}, b, \xi_1, \dots, \xi_n, \xi_1^*, \dots, \xi_k^*); \\ & \frac{1}{2} \|\bar{w}\|^2 + C \sum_{i=0}^n \xi_i + C^* \sum_{j=0}^k \xi_j^* \\ & \text{subject to:} \\ & \forall_{i=1}^n : y_i [\bar{w} \cdot \bar{x}_i + b] \geq 1 - \xi_i \\ & \forall_{j=1}^k : y_j^* [\bar{w} \cdot \bar{x}_j^* + b] \geq 1 - \xi_j^* \\ & \forall_{i=1}^n : \xi_i > 0 \\ & \forall_{j=1}^k : \xi_j^* > 0 \end{aligned} \right\} \tag{16}$$

C and C^* are parameters set by the user. They allow trading off margin size against misclassifying training examples or excluding test examples.

Standard support vector machines produce an uncalibrated value that is not a probability. But constructing a classifier to produce a posterior probability is very useful, for example, posterior probabilities are required when a classifier is making a small part of an overall decision, and the classification outputs must be combined for the overall decision. In this paper, we adopted a function[5] to map the SVM outputs into probabilities. Instead of estimating the class-conditional densities $p(f | y)$, a parametric model is used to fit the posterior $P(y = 1 | f)$ directly.

$$P(y = 1 | f) = \frac{1}{1 + e^{-f(x)}} \quad (17)$$

4 Experiment Results

The dataset used in event detection is NEM study Corpus[6]. The NEM Study Corpus comprises a set of Internet news stories, which are collected from several influential websites. Each story is represented as a stream of text. It consists of 16560 chronologically ordered news stories. There are 16 events manually identified in this corpus. Each story was assigned a label of “Yes”, “No” or “Brief” with respect to each of the 16 events. Except for a small fraction of the articles, each document belongs to exactly one event. We choose 10 events from NEM study corpus to carry out event detection and tracking experiment.

The effectiveness measures for event detection is calculated using the 4 element based contingency table^[1]. To evaluate the effectiveness of the event tracking results, six effectiveness measures including Recall, Precision, F1-measure, P_{miss} , P_{false} and $Cost_{norm}$ are used[7].

Table 1 shows the measure results of IKM-based detection system by using different effective density radiuses. We observe that an appropriate destination can be reached if we let $R_d=2R_m$, because not only $Cost_{norm}$ is smallest, but also each event follow the same rule.

Table 1. Effectiveness measures of different effective density radiuses

Density radius	Recall	Precision	P_{miss}	P_{false}	$Cost_{norm}$
$R_d=R_m/8$	90.83%	83.21%	9.17%	2.04%	0.1917
$R_d=R_m/6$	90.83%	83.21%	9.17%	2.04%	0.1917
$R_d=R_m/4$	89.17%	80.86%	10.83%	2.35%	0.2235
$R_d=R_m/2$	82.78%	51.56%	17.22%	8.64%	0.5956
$R_d=R_m$	86.11%	74.52%	13.89%	3.27%	0.2991
$R_d=2R_m$	97.50%	97.50%	2.5%	0.28%	0.0387
$R_d=2.5R_m$	96.94%	86.39%	3.06%	1.70%	0.1139
$R_d=3R_m$	93.06%	66.21%	6.94%	5.28%	0.3281

In order to compare with other event detection methods, we adopt P_{miss} , P_{false} and $Cost_{norm}$ to evaluate the performance of system. Table 2 shows the comparison of different event detection methods($T_{Dis\ tan\ ce} = 1.70$).

Table 2. Comparison of different event detection methods

Method	P_{miss}	P_{false}	$\text{Cost}_{\text{norm}}$
Single-pass	9.02%	0.68%	0.1235
T-Kmeans	3.35%	0.30%	0.0482
IIKM	1.67%	0.19%	0.0260

We can observe that the IIKM based event detection system showed a better performance than other event detection methods. Miss rate, false alarm rate and normalized cost of IKM based system are lower than those of other system. The normalized cost of IIKM is reduced by 0.0975 and 0.0222 respectively, when compared to those of Single-pass, Traditional k-means and IIKM.

Table 3. Comparison of different event tracking methods

Method	P_{miss}	P_{false}	$\text{Cost}_{\text{norm}}$
Rocchio	8.0788%	0.4306%	0.1019
D-Trees	2.0395%	1.3307%	0.0856
KNN	0.9621%	0.4580%	0.0321
TSVM	0.9741%	0.4229%	0.0304

Table 3 shows a result from comparison of four tracking algorithms. As it can be seen from the result, our tracking method based on TSVM with probabilistic outputs showed a better performance than other methods, the normalized cost of TSVM is reduced by 0.0715, 0.0552 and 0.0017 respectively, when compared to those of Rocchio, Decision Trees and KNN. Though KNN is gaining popularity due to many attractive features and promising performance in the fields of event tracking, the performance of our algorithm is close to that of KNN in miss rate, but as far as the normalized cost is concerned, TSVM performs better than KNN.

5 Conclusions

In this paper, we introduced an IIKM based algorithm to detect news event. The algorithm utilizes density function to initialize cluster centers to select initial cluster centers objectively, and the quantity of clusters is affected little by the order in which the news stories are processed. Furthermore, we have put forward a TSVM based algorithm to track topic relevant event. It is especially fit for event tracking when the training set is small. In order to offer a confidence measure the TSVM outputs are mapped into probabilities. With real world data sets, experimental results indicate that the methods proposed by this paper are effective.

In our future work we plan to concentrate on exploring the suitability of information extraction techniques to event detection and tracking, and their effects on system performance. We believe the results are encouraging for the pursuit of some novel strategies of event detection and tracking.

References

1. Allan, J., Carbonell, J., Doddington, G., et al.: Topic Detection and Tracking Pilot Study Final Report. In: Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop, pp. 194–218. Morgan Kaufmann Publishers, San Francisco (1998)
2. Lei, Z., Wu, L.D., Lei, L., Huang, Y.Y.: Incremental K-means Method Base on Initialisation of Cluster Centers and Its application in News Event Detection. *Journal of the China Society for Scientific and Technical Information* 25(3), 289–295 (2006)
3. Joachims, T.: Transductive Inference for Text Classification Using Support Vector Machines. In: Proceedings of the 16th International Conference on Machine Learning (ICML), pp. 200–209. Morgan Kaufmann Publishers, San Francisco (1999)
4. Wahba, G.: Support Vector Machines, Reproducing Kernel Hilbert Spaces and the Randomized GACV. In: *Advances in Kernel Methods Support Vector Learning*, pp. 69–88. MIT Press, Massachusetts (1999)
5. Pei, J.H., Fan, J.L., Xie, W.X.: The Initialization Method of Cluster Centers. *Journal of Electronics* 21(3), 320–325 (1999)
6. Lei, Z., Wu, L.D., Lei, L., Liu, Y.C.: A System for Event Detection and Tracking Based on Constructive-Competition Clustering and KNNFL. *System Engineering Theory and Practice* 26(3), 68–74 (2006)
7. Papka, R.: On-line New Event Detection, Clustering, and Tracking. Ph.D. Thesis, University of Massachusetts at Amherst (1999)

Integration of Fuzzy Logic in Data Mining to Handle Vagueness and Uncertainty

Raju G.¹, Binu Thomas², Th. Shanta Kumar³, and Sangay Thinley²

¹ SCMS School of Technology & Management, Cochin, Kerala, India

kurupgraju@rediffmail.com

² Department of Mathematics & Computer Science, Sherubtse College, Bhutan

binumarian@rediffmail.com

³ Research scholar, Himachal Pradesh University, India

Abstract. Recent developments in the fields of business investment, scientific research and information technology have resulted in the collection of massive data which becomes highly useful in finding certain patterns governing the data source. Clustering algorithms are popular in finding hidden patterns and information from such repository of data. The conventional clustering algorithms have difficulties in handling the challenges posed by the collection of natural data which is often vague and uncertain. This paper presents the concept of fuzzy clustering (fuzzy c-means clustering) and shows how it can handle vagueness and uncertainty in comparison with the conventional k-means clustering algorithm.

Keywords: Clustering, data mining, fuzzy c-means, fuzzy clustering, k-means, overlapping clusters, vagueness and uncertainty.

1 Introduction

Data Mining or Knowledge discovery refers to a variety of techniques that have developed in the fields of databases, machine learning and pattern recognition [1]. The intent of these techniques is to uncover useful patterns and associations from large databases. Large databases in commerce driven industries such as retailing, manufacturing, insurance, banking and investments contain a staggering amount of raw data. Knowledge engineers and system analysts are now faced with challenges of maintaining a competitive edge in the world of e-commerce and rapidly changing delivery channels. To meet these challenges, they must look into the accumulated raw data to find new customer relationships, possible newly emerging lines of business and ways to cross sell existing products to new and existing customers [10]. The process of finding useful patterns and information from raw data is often known as Knowledge discovery in databases or KDD. Data mining is a particular step in this process involving the application of specific algorithms for extracting patterns (models) from data [5].

Cluster analysis is a technique for breaking data down into related components in such a way that patterns and order becomes visible. It aims at sifting through large volumes of data in order to reveal useful information in the form of new

relationships, patterns, or clusters, for decision-making by a user. Clusters are natural groupings of data items based on similarity metrics or probability density models. Clustering algorithms maps a new data item into one of several known clusters. In fact cluster analysis has the virtue of strengthening the exposure of patterns and behavior as more and more data becomes available [7]. A cluster has a center of gravity which is basically the weighted average of the cluster. Membership of a data item in a cluster can be determined by measuring the distance from each cluster center to the data point [6]. The data item is added to a cluster for which this distance is a minimum.

This paper provides an overview of the crisp clustering technique, its limitation in handling natural data and the ability of fuzzy clustering method to process natural data with vagueness and uncertainty. Section 2 describes the basic notions of clustering and also introduces k-means clustering algorithm. In Section 3 we explain the concept of vagueness and uncertainty in natural data. Section 4 introduces the fuzzy clustering algorithm and describes how it can handle vagueness and uncertainty with the concept of overlapping clusters with partial membership functions. This section also explains the fuzzy c-means algorithm. Section 5 demonstrates the concepts presented in the paper. Finally, section 6 concludes the paper.

2 Crisp Clustering Techniques

Traditional clustering techniques attempt to segment data by grouping related attributes in uniquely defined clusters. Each data point in the sample space is assigned to only one cluster. In partitioning the data only cluster centers are moved and none of the data points are moved [12]. Thus clustering is an iterative process of finding better and better cluster centers in each step.

K-means algorithm and its different variations are the most well-known and commonly used partitioning methods. The value 'k' stands for the number of cluster seeds initially provided for the algorithm. This algorithm takes the input parameter 'k' and partitions a set of m objects into k clusters [7]. The technique work by computing the distance between a data point and the cluster center to add an item into one of the clusters so that intra-cluster similarity is high but inter-cluster similarity is low. A common method to find the distance is to calculate to sum of the squared difference as follows and it is known as the Euclidian distance [10](exp.1).

$$d_k = \sum_n \left\| X_j^k - C_i \right\|^2 \quad (1)$$

where,

- d_k : is the distance of the k^{th} data point
- n : is the number of attributes in a cluster
- X_j^k : is j^{th} value of the k^{th} data point
- C_j^i : is the j^{th} value of the i^{th} cluster center

Pseudo code of k-means clustering algorithm [10] is given below:

```

initialise k=number of clusters
initialise  $C_j$  (cluster centers)
Set Cycle variable  $t=1$ 
Repeat
  For  $i=1$  to  $n$  Distribute sample points( $x_i$ ) into  $k$  clusters
  For  $j=1$  to  $k$  : Calculate  $S_j^{(t)}$  for  $x_i$  applying (2)
  For  $j=1$  to  $k$  : Compute new cluster centers by calculating weighted average
   $t=t+1$ 
Until  $C_j$  estimate stabilize

```

With the definition of the distance of a data point from the cluster centers, the k-means algorithm is fairly simple. The cluster centers are randomly initialized and we assign data points x_i into clusters based on their distance to cluster centers using expression (2).

$$S_j^{(t)} = \{ \|x_i - C_j^{(t)}\| < \|x_i - C_k^{(t)}\|, j \neq k \} \quad (2)$$

Here

$S_j^{(t)}$: is the membership of x_i in j^{th} cluster
 t : is the current iteration cycle value
 x_i : is the i^{th} data point
 C_j : is the center of the j^{th} cluster
 C_k : is the center of the k^{th} cluster

When all the data points have been assigned to clusters, new cluster centers are calculated. The cluster center calculation causes the previous centroid location to move towards the center of the cluster set. This is continued until there is no change in cluster centers.

3 Uncertainties and Vagueness in Real World Data

The real world data is almost never arranged in such clear cut groups. Instead, clusters have ill defined boundaries that smear into the data space often overlapping the perimeters of surrounding clusters [4]. In most of the cases the real world data have apparent extraneous data points clearly not belonging to any of the clusters. This happens because natural data do not happen to appear in clear-cut crisp fashion but suffers from the following limitations:

- 1) Not clearly known : Questionable; problematical
- 2) Vague : Not definite of determined
- 3) Doubtful : Not having certain information
- 4) Ambiguous : Many interpretations
- 5) Not steady : Varying
- 6) Liable to change : Not dependable or reliable

When we further examine these meanings two categories of limitations emerge quite naturally – Uncertainty and Vagueness [8]. Vagueness is associated with the difficulty of making sharp and precise distinctions in the world. Uncertainty is a situation in which the choice between two or more alternatives is left unspecified [11]. The modeling of imprecise and qualitative knowledge, as well as handling of uncertainty at various stages is possible through the use of fuzzy sets. Fuzzy logic is capable of supporting, to a reasonable extent, human type reasoning in natural form by allowing partial membership for data items in fuzzy subsets [2]. Integration of fuzzy logic with data mining techniques has become one of the key constituents of soft computing in handling the challenges posed by the massive collection of natural data [1].

4 Fuzzy Clustering Algorithm

The central idea in fuzzy clustering is the non-unique partitioning of the data in a collection of clusters. The data points are assigned membership values for each of the clusters. The fuzzy clustering algorithms allow the clusters to grow into their natural shapes [15]. In some cases the membership value may be zero indicating that the data point is not a member of the cluster under consideration. Many crisp clustering techniques have difficulties in handling extreme outliers but fuzzy clustering algorithms tend to give them very small membership degree in surrounding clusters [14]. The non-zero membership values, with a maximum of one, show the degree to which the data point represents a cluster. Thus fuzzy clustering provides a flexible and robust method for handling natural data with vagueness and uncertainty. In fuzzy clustering, each data point will have an associated degree of membership for each cluster. The membership value is in the range zero to one and indicates the strength of its association in that cluster.

Fuzzy c-means clustering involves two processes: the calculation of cluster centers and the assignment of points to these centers using a form of Euclidian distance. This process is repeated until the cluster centers stabilize. The algorithm is similar to k-means clustering in many ways but it assigns a membership value to the data items for the clusters with in a range of 0 to 1. So it incorporates fuzzy set's concepts of partial membership and forms overlapping clusters to support it. The algorithm needs a fuzzification parameter m in the range $[1, n]$ which determines the degree of fuzziness in the clusters. When m reaches the value of 1 the algorithm works like a crisp partitioning algorithm and for larger values of m the overlapping of clusters is tend to be more. The algorithm calculates the membership value μ with the formula,

$$\mu_j(x_i) = \frac{\left(\frac{1}{d_{ji}}\right)^{\frac{1}{m-1}}}{\sum_{k=1}^p \left(\frac{1}{d_{ki}}\right)^{\frac{1}{m-1}}} \quad (3)$$

where

- $\mu_j(x_i)$: is the membership of x_i in the j^{th} cluster
- d_{ji} : is the distance of x_i in cluster c_j
- m : is the fuzzification parameter
- p : is the number of specified clusters
- d_{ki} : is the distance of x_i in cluster C_k

The new cluster centers are calculated with these membership values using the exp. 4

$$c_j = \frac{\sum_i [\mu_j(x_i)]^m x_i}{\sum_i [\mu_j(x_i)]^m} \quad (4)$$

where

- C_j : is the center of the j^{th} cluster
- x_i : is the i^{th} data point
- μ_j : the function which returns the membership
- m : is the fuzzification parameter

This is a special form of weighted average. We modify the degree of fuzzyness in x_i 's current membership and multiply this by x_i . The product obtained is divided by the sum of the fuzzified membership.

Pseudo code of fuzzy c-means clustering algorithm [10] is given below:

```

initialize p=number of clusters
initialize m=fuzzification parameter
initialize  $C_j$  (cluster centers)
Repeat
  For i=1 to n :Update  $\mu_j(x_i)$  applying (3)
  For j=1 to p :Update  $C_i$  with (4) with current  $\mu_j(x_i)$ 
Until  $C_j$  estimate stabilize

```

The first loop of the algorithm calculates membership values for the data points in clusters and the second loop recalculates the cluster centers using these membership values. When the cluster center stabilizes (when there is no change) the algorithm ends. Thus the fuzzy c-means clustering algorithm is efficient in handling data with outlier points or natural data with uncertainty and vagueness. It accomplishes this by virtue of its unique partial membership features for data items in different clusters so that the clusters grow naturally to reveal hidden patterns.

5 Illustration

Consider the monthly income and expenses of four different people given in table 1.

Table 1. Monthly income, expenses and memberships in cluster centers C1=P1 and C2=P2

Person (Points)	Income in 1000s	Expenses in 1000s	Distance From C1=P1	Distance From C2=P2	Membership in
P1	20	4	0	20.1	C1
P2	40	6	20.1	0	C2
P3	60	10	40.45	20.4	C2
P4	80	12	60.53	40.45	C2

An illustration of how the two algorithms works is discussed below:

K-means clustering algorithm can be applied on the above data points with initial values of $k=2, C1=P1$ and $C2=P2$ to identify the cluster memberships (Table1) using expressions (1) and (2). New cluster centers are found by calculating the weighted average of all data points in a cluster (figure 1). This process is repeated until the algorithm stabilizes to the situation represented in figure 2, with cluster centers at $C1(30,5)$ and $C2(70,11)$.

Now one more data point $P5(50,8)$ is added to the set (fig. 3). The Euclidean distances of P5 from both $C1(30,5)$ and $C2(70,11)$ are found to be 20.22. This leads to an uncertain situation and the k-means clustering algorithm fails to include this point to any one of the existing clusters.

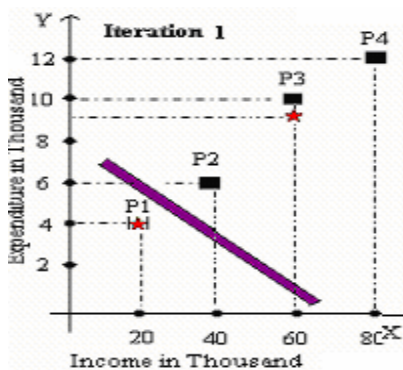


Fig. 1. New cluster centers(stars) after the first iteration of k-means algorithm

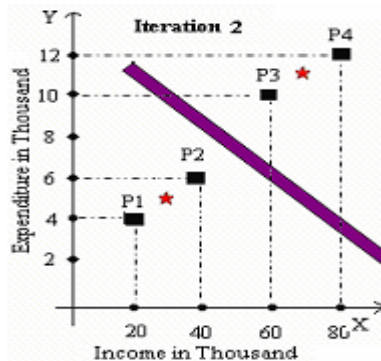


Fig. 2. The final stabilized situation with cluster centers at $C1(30,5)$ and $C2(70,11)$

To this modified data, the fuzzy c-means algorithm is applied, with initial values $m=2$ and $p=2$. The algorithm finds membership values for each data points in two cluster centers using exp.3. This membership value reflects the degree to which the point is more representative of one cluster than another.

Table 2. Fuzzy membership values of 5 points in two cluster centers C1 & C2 using exp. 3

Points	Distance from C1(30,5)	Distance from C2(70,11)	Memberships in C1	Memberships in C2
P1(20,4)	10.05	50.488	.834	.166
P2(40,6)	10.05	30.414	.7516	.2484
P3(60,10)	30.414	10.05	.2484	.7516
P4(80,12)	50.488	10.05	.166	.8340
P5(50,8)	20.224	20.224	.5	.5

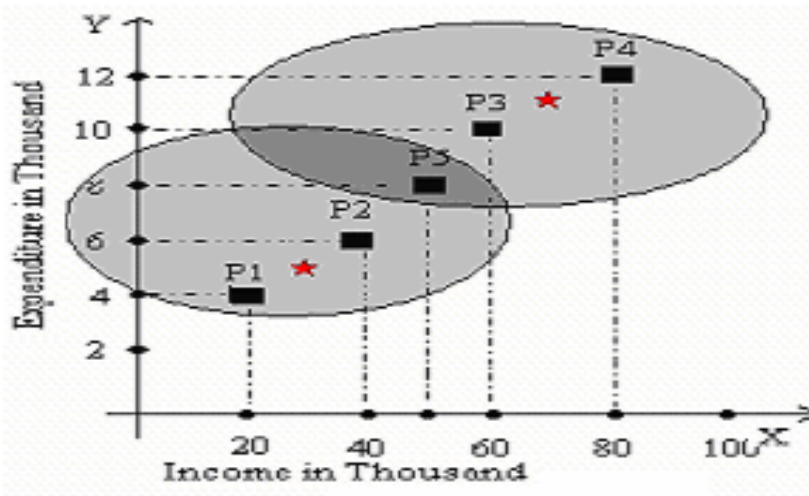


Fig.3 Fuzzy overlapping clusters with partial membership of .5 for P5 in C1 and C2

6 Conclusion

A good clustering algorithm produces high quality clusters to yield low inter cluster similarity and high intra cluster similarity. Many conventional clustering algorithms like k-means clustering algorithm achieve this on crisp and highly structured data. But they have difficulties in handling unstructured natural data which often contain outlier data points. Fuzzy sets and fuzzy clustering algorithms have a significant role to play in handling the natural data which suffer from vagueness and uncertainty. The fuzzy c-means algorithm is an improved version of the k-means algorithm which can handle vague and uncertain data by assigning partial memberships to the data points in different clusters. The fuzzy algorithms are found to be more efficient in obtaining hidden patterns and information from massive collection of natural data, which usually poses challenges to the conventional clustering algorithms.

References

1. Mitra, S., Pal, S.K., Mitra, P.: Data Mining in Soft Computing Framework: A Survey. *IEEE Trans. on Neural Networks* 13(1), 3–14 (2002)
2. Cruse, R., Borgelt, C.: Fuzzy Data Analysis Challenges and Perspective, <http://citeseer.ist.psu.edu/kruse99fuzzy.html>
3. Au, W.H., Keith, C.C.: Classification with Degree of Membership: A Fuzzy Approach. In: *Proc. IEEE International Conf on Data Mining (ICDM 2001)*, pp. 35–42 (2001)
4. Halkidi, M.: Quality Assessment and Uncertainty Handling in Data Mining Process, <http://citeseer.ist.psu.edu/halkidi00quality.html>
5. Inmon, W.H.: The Data Warehouse and Data Mining. *Comm. Of ACM* 39(11), 49–50 (1996)
6. Fayyad, U., Uthurusamy, R.: Data Mining and Knowledge Discovery in Databases. *Comm. of ACM* 39(11), 24–26 (1996)
7. Berkhin, P.: Survey of Clustering Data Mining Techniques, <http://citeseer.ist.psu.edu/berkhin02survey.html>
8. Chau, M., Cheng, R., Kao, B.: Uncertain Data Mining: A New Research Direction, http://www.business.hku.hk/~mchau/papers/UncertainDataMining_WSA.pdf
9. Keith, C.C., Au, W.H., Choi, B.: Mining Fuzzy Rules in a Donor Database for Direct Marketing by a Charitable Organization. In: *Proc. of 1st IEEE International Conference on Cognitive Informatics*, pp. 239–246 (2002)
10. Cox, E.: *Fuzzy Modeling and Genetic Algorithms for Data Mining and Exploration*. Elsevier, Amsterdam (2005)
11. Klir, G.J., Folger, T.A.: *Fuzzy Sets, Uncertainty and Information*. Prentice Hall, Englewood Cliffs (1988)
12. Han, J., Kamber, M.: *Data Mining Concepts and Techniques*. Elsevier, Amsterdam (2003)
13. Bezdek, J.C.: *Fuzzy Mathematics in Pattern Classification*, Ph.D. Thesis, Center for Applied Mathematics, Cornell University, Ithica N.Y (1973)
14. Looney, C.G.: A Fuzzy Clustering and Fuzzy Merging Algorithm, <http://citeseer.ist.psu.edu/399498.html>
15. Klawonn, F., Keller, A.: Fuzzy Clustering Based on Modified Distance Measures, <http://citeseer.ist.psu.edu/klawonn99fuzzy.html>

Medical Knowledge Discovery from a Regional Asthma Dataset

Sam Schmidt, Gang Li, and Yi-Ping Phoebe Chen

School of Engineering and Information Technology, Deakin University
221 Burwood Highway, Vic 3125, Australia
sam.schmidt@deakin.edu.au, gang.li@deakin.edu.au,
phoebe.chen@deakin.edu.au

Abstract. Paediatric asthma represents a significant public health problem. To date, clinical data sets have typically been examined using traditional data analysis techniques. While such traditional statistical methods are invariably widespread, large volumes of data may overwhelm such approaches. The new generation of knowledge discovery techniques may therefore be a more appropriate means of analysis. The primary purpose of this study was to investigate an asthma data set, with the application of various data mining techniques for knowledge discovery. The current study utilises data from an asthma data set ($n \approx 17000$). The findings revealed a number of factors and patterns of interest.

1 Introduction and Background

Asthma is classed as a chronic incurable disease, with the World Health Organisation classifying the disease as ‘recurrent attacks of breathlessness and wheezing’ [1]. In Australia, recent surveys suggest that the prevalence of asthma in school-aged children is amongst the highest in the world, with $\approx 14 - 16\%$ of children being diagnosed with the disease [1]. In addition, the disease is responsible for one of the highest instances of hospitalisation among children, and hence is associated with a significant economic cost burden to the Australian community, a financial equivalence reported at 4.3 billion AUD [2]. Therefore, it is not surprising that health care systems are seeking technological solutions and strategies that may contribute to the successful treatment and prevention of asthma. Despite this, the application of knowledge discovery to biomedical data in relation to asthma is an area that to date has lacked attention.

Computational techniques and information technologies permit the management and examination of large sets of biomedical data. Specifically, the application of knowledge discovery is an area adopted by various successful biomedical applications [3]. Fayyad et al [4] defines knowledge discovery as ‘the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data’. Knowledge discovery is increasingly being acknowledged as having significant relevance to medical areas. For example, Kudyba [5] states

¹ W.H.O. <<http://www.who.int/respiratory/asthma/en/>>

that ‘strong disease management programs depend on data mining techniques’. More recently, Krieger et al [6] outlined the future trends in data mining, and identified the issue of increasing usability by making data mining methods more user friendly, such techniques have the potential to further strengthen ties between computer science and the medical profession. However, such a task becomes difficult when barriers are introduced with heterogenous medical data. The ‘uniqueness’ of medical data mining, as described by Moore and Krzysztof [7] may be categorically defined into four groups: *Heterogeneity of medical data*, *Ethical, legal, and social issues*, *Statistical philosophy* and *Special status of medicine*. These four groups highlight the differences and challenges that can exist in medical data mining, perhaps offering an explanation as to why the application of data mining in medical science is still limited.

The current study intends to apply knowledge discovery techniques to search for relationships and patterns within a large existing data set ($n = 16976$) that may otherwise be obscured by solely using traditional statistical methods of data analysis.

Data was collected between March and September 2005 in Victoria, Australia, by sampling a random group of primary schools [8]. A total of 92 distinct schools were surveyed, resulting in a dataset of ≈ 17000 records. Data matching with the Australian Bureau of Statistics Census survey was completed, deriving new variables: *socio-economic*, *regional* and *industrial* indices. These methodologies are described in detail elsewhere [8]. While previous analyses on this data set have utilised traditional statistical approaches, the current study extends research in this area by moving to the application of knowledge discovery employing various data mining techniques.

2 Data Preparation and Transformation

This study follows the Cross Industry Standard Practice for Data Mining (CRISP-DM) [9] methodology, however a deviation from the specification was required: data visualisation in regard to data structure. Specifically, within the ‘data understanding’ phase data structural visualisation is not included. In this study the ‘explore data’ process was exploited to visualise the data structure rather than local attributes (or variables). It may perhaps be vital for comprehensibility across industries to broaden the scope of this phase.

After general understanding of the data set, the next step was to conduct data cleaning and preparation. As aforementioned, a graph comprehensible across professions was created, drawing from a set of principles under the acronym ACCENT with the aim of maximising the visual effectiveness of data [10]. This is founded upon six principles: Apprehension, Clarity, Consistency, Efficiency, Necessity and Truthfulness. The visualisation of the data structure proved to be extremely valuable, enabling instant viewing and recognition of the data set in its entirety. The data visualisation was constructed in *DOT* language and processed with *Graph Viz* to create a hierarchical tree diagram. Furthermore, embedding metadata could greatly enhance knowledge communication. For example, the

Table 1. Variables in newly transformed asthma data

Demographics		Symptoms		Asthma		Medication/Management	
Variable	Description	Variable	Description	Variable	Description	Variable	Description
gender	Child gender	evwhz_r	Ever wheezed	drforast	Doctor for asthma	antichol1_c	Anticholinergics
age	Child age	evwhz12m_r	Wheezed within 12 months	drforwhz	Doctor visits due to wheeze	betagl_c	Beta agonist
seifa	Socio-economic Index	exwhz_r	Wheeze during exercise	emerg	Visits to hospital emergency	inhster1_c	Inhaled steroid
absregion	Region code	ntcough_r	Coughing during night	everast_r	Reported asthma	leukant_c	Leukotriene antagonist
industriala	Industrial area	sleepdisturb	Sleep disturbance	hospadm	Hospital admissions	pipafreq_c	Parent-initiated Prednisolone

interpretation between professions may vary, however, the visualisation provides an instant means for recognition of the data structure.

The data set was then grouped into the following categories: Demographics (general demographics); Symptoms (known asthma symptoms); Asthma (confirmed instances of asthma); and Medication and Management (medication types and treatment). The data set initially contained 77 attributes, later optimised into ≈ 30 attributes after data preparation and cleaning. Table 1 lists the 20 variables under examination in the current study.

3 Knowledge Discovery

Initially, segmentation was used for visualisation and comprehensibility of the data. Association was then performed, followed by classification and neural networks.

3.1 Segmentation

To assist with data comprehensibility, Kohonen’s Self-Organising Maps (SOM) were used to segment data and associate different attributes. This study examined in particular a set of attributes relating to children with asthma with the intention to discover new knowledge via visual patterns in the SOMs. This was achieved using the following equation devised by Kohonen [11]:

$$\Delta \mathbf{w}_i = c_i(\mathbf{x} - \mathbf{w}_i^{old}), \text{ for } i \in N_r \tag{1}$$

As described in equation 1, each node has a weight vector $\Delta \mathbf{w}_i$, initially having a random weighting; this changes through the learning process as the euclidean distance is calculated between \mathbf{x} and \mathbf{w}_i and the smallest distance is chosen. The neighbours are then updated. c_i represents the learning coefficient to weigh the node to result in the node being more distant from others. This process iterates until all nodes are stabilised, and neighbours have the most similar weighting values.

Figure 1 presents an example of six SOM’s, associated with the variable *sleep-disturb*. Comprehensibility is demonstrated, and many visual conclusions may be drawn. For instance, examination of figure 1 from top left to bottom right indicates that:

1. Children who are least-disadvantaged in terms of the socio-economic indice (*seifa*) are most likely to have sleep disturbance.
2. Younger children ($age \leq 8$) are most likely to have a sleep disturbance.

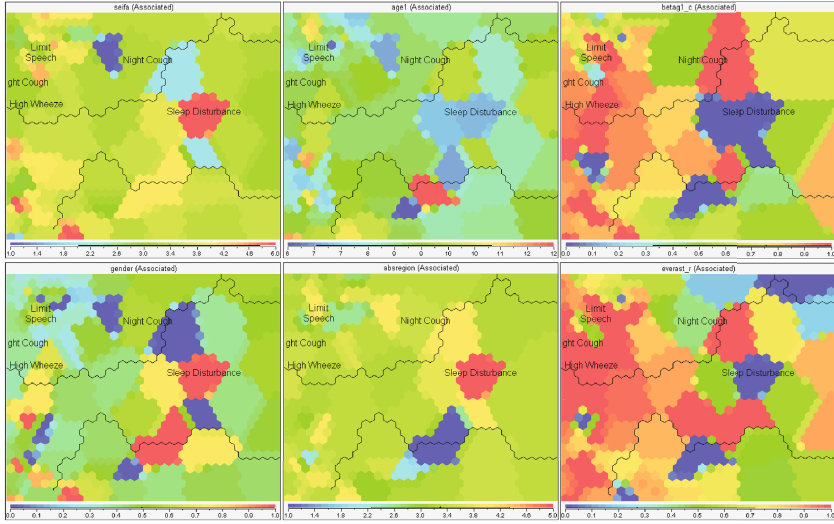


Fig. 1. SOM's in relation to sleep disturbance in children

3. Children with reported asthma on medication (*betag1_c*), are the least likely to have sleep disturbance.
4. Females are much more likely to have sleep disturbed than males.
5. The region (*absregion*) most likely to have sleep disturbance is region 5, which represents children living in an 'Inner City' area.
6. Children with reported asthma have a low instance of sleep disturbance (*sleepdisturb*).

From these results, further logical conclusions and knowledge discovery can be achieved; the findings provide preliminary evidence to suggest that children living in an inner city area are prone to more disturbances, perhaps due to general city activity during the night, e.g. police sirens, alarms, people. In contrast, children living in rural areas may not be subject to these disturbances.

Likewise, the correlation between the socio-economic index shows that the most advantaged group of children have high sleep disturbance, which directly correlates with the child's region; i.e. the highest socio-economic value is located in 'inner city' areas. Therefore, the conclusion that highly advantaged children suffer sleep disturbance due to asthma is unlikely, and the weighting of this asthma marker may be lowered. As a result, whilst association algorithms such as *apriori* rate sleep disturbance as a highly confident attribute to predict whether a child has asthma, it may not be the best attribute to use in asthma prediction.

Additionally, these results suggest that further research may be necessary to investigate the high prevalence of sleep disturbance found in females, in comparison to males. The result may then be used to weight various attributes for a more precise prediction of asthma instances. Likewise, with younger children being more likely to have sleep disturbance, further investigation is required. Fur-

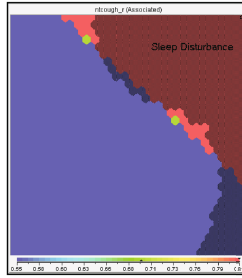


Fig. 2. Sleep disturbance and night cough correlation

thermore, the investigation of sleep disturbance against single attributes, derived from an initial comparison against all attributes produced visually identifying patterns of high correlation. Figure 2 presents the results of children with a sleep disturbance, as indicated by the red colouring, top right of the SOM. The shaded area overlapping much of this same area represents the attribute *ntcough_r*. Examination of figure 2 suggests that there is a high correlation between these two asthma markers. One possible explanation for this correlation is that if a child is coughing at night, their sleep may be disturbed, and vice versa. Such implicit knowledge discovery may be highly beneficial in the design of future research instruments. For example, these findings suggest that the accuracy of such questions may be skewed and ideally the two original questions could be merged into a single question to achieve more reliable findings.

3.2 Association

The *apriori* algorithm was initially used for knowledge discovery to discover associations with confirmed asthma cases in children. The *apriori* algorithm was initially developed for speed and use on ‘basket data’ [12]. In this example, due to the speed and robustness of the *apriori* algorithm it is applied to the set of asthma symptoms as collected in the initial research instrument. The algorithm produced fifteen rules; the resulting consequents and antecedents with support and confidence percentages are presented in table 2.

From the application of the *apriori* algorithm, the discovery of children with confirmed asthma, also shows high confidence that they will have most likely

Table 2. Ten rules derived from *apriori* association

Consequent	Antecedent	Support	Confidence
<i>everast_r</i> = 1	<i>exwhz_r</i> = 1 and <i>ntcough_r</i> = 1 and <i>evwhz_r</i> = 1	10.434	92.557
<i>everast_r</i> = 1	<i>exwhz_r</i> = 1 and <i>evwhz12m_r</i> = 1 and <i>evwhz_r</i> = 1	12.41	91.202
<i>everast_r</i> = 1	<i>exwhz_r</i> = 1 and <i>evwhz12m_r</i> = 1	12.433	91.156
<i>everast_r</i> = 1	<i>exwhz_r</i> = 1 and <i>ntcough_r</i> = 1	10.818	90.903
<i>everast_r</i> = 1	<i>exwhz_r</i> = 1 and <i>evwhz_r</i> = 1	13.525	90.733
<i>everast_r</i> = 1	<i>exwhz_r</i> = 1	14.14	88.309
<i>everast_r</i> = 1	<i>ntcough_r</i> = 1 and <i>evwhz12m_r</i> = 1 and <i>evwhz_r</i> = 1	13.501	87.528
<i>everast_r</i> = 1	<i>ntcough_r</i> = 1 and <i>evwhz12m_r</i> = 1	13.54	87.507
<i>everast_r</i> = 1	<i>sleepdisturb</i> = 1 and <i>evwhz_r</i> = 1	10.872	86.634
<i>everast_r</i> = 1	<i>sleepdisturb</i> = 1 and <i>evwhz12m_r</i> = 1 and <i>evwhz_r</i> = 1	10.872	86.634

wheezed in the past, wheezed during exercise, and had a night cough. Further variables in order of confidence show exercise wheeze, ever wheezed and wheezed in the past twelve months and so forth. Such results not only assist in data understanding, but may be used to validate other algorithms such as the *C5.0* decision tree algorithm as discussed in the next section.

3.3 Classification and Neural Networks

The *C5.0* Algorithm was used to produce a decision tree, based on the likelihood of a child getting asthma given the symptomatic variables contained in the data set. The resulting output of the *C5.0* decision tree was converted to a graphical format using *DOT* language for added comprehensibility, as shown in figure 3.

Many similar algorithms were performed for comparison in accuracy and comprehensibility. The examples include *Chi-Squared Automatic Interaction Detector* (CHAID), the *Quick, Unbiased and Efficient Statistical Tree* (QUEST) and other such binary split decision tree algorithms. Likewise, *C4.5* and *C5.0* were also compared and found to produce almost identical results, however *C5.0* was faster in operation. *C5.0* was chosen not only for speed, but the ability to accurately prune the tree to a desired level and the resulting detail for asthma predication.

The resulting *C5.0* algorithm was created using 80/20 of the dataset, 80% of the dataset for training and the remaining 20% of the dataset for testing and evaluating the accuracy of the tree. The overall accuracy resulted at 88.2%. With aggressive pruning, the tree could be simplified to a single node containing

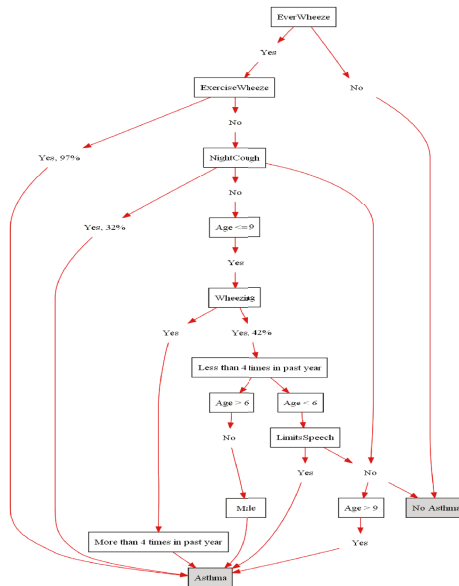


Fig. 3. Decision tree for childhood asthma

the attribute ‘ever wheezed’ to produce an accuracy of 92%. Whilst the latter accuracy is higher than the model this study used, it is however not as detailed and therefore cannot be used as an efficient tool for predication. To compare with the decision tree classification methods, a neural network was also developed. Using an exhaustive prune model, the same 12 inputs were introduced to predict if a child will have asthma. Again, an 80/20 split was used. The estimated accuracy of the neural network achieved above 90% using the same inputs as per the classification tree. Within the neural network, surprisingly, the well known correlation between gender and asthma was not highly deterministic of a child having asthma, with gender being rated as the lowest determining factor at 7%. This finding is not consistent with existing research for children of this age group which has found the opposite to be true [1]. Likewise, ‘children living in industrial areas’ was surprisingly the second weakest indicator at 9%, when research has consistently demonstrated that air pollutants are common asthma triggers [1].

4 Conclusions

This research study demonstrates that the use of knowledge discovery techniques such as data mining may uncover previously unknown correlations and new issues relating to the characteristics asthma. This has vast implications when studying the epidemiology of any disease or condition. Furthermore, such insights may be of great economic value at an international level, as health systems across the globe struggle with the cost burden associated with asthma.

Such analysis holds enormous potential to unlock useful patterns and knowledge which may otherwise be obscured [3]. Through the knowledge discovery process, in particular the use of Kohonen’s SOM’s, a variety of correlations were identified. Most importantly, knowledge discovery of seemingly implicit attributes were identified and shown to be sub-optimal in the diagnosis of asthma. For example, the finding that sleep disturbance and coughing at night have a high correlation, and implicitly and logically relate to one another. It is therefore recommended that future studies consider this association when designing research instruments.

Findings from the decision tree results discovered by the *C5.0* algorithm (see figure 3) did not support existing research under some branches. Further research is therefore required to examine whether this finding is atypical. The findings from the current study indicate that the utilisation of new knowledge discovery techniques in medical science areas should be encouraged. Co-operation between the two professions would also increase understanding of the aims and abilities of either side to forge new understandings and knowledge. Such development could have global significance.

The presence of computer science in the medical realm is increasing, with the potential for far reaching ramifications, particularly in the areas of epidemiology, prevention and public health overall. Professionals in these fields may have different training and knowledge, and such differences may foster scientific

miscommunication. As a consequence, the possible implications of such differences between fields in the research setting requires consideration.

Acknowledgements. Thank you to Dr. Peter Vuillermin and Dr. Bernie Jenner, consultant paediatricians with specialisation in childhood asthma, associate paediatricians to the Royal Children's Hospital (Melbourne) and lecturers at Melbourne University. The related work is supported by Deakin CRGS Grant 2008.

References

1. Australian Government: Chronic Respiratory Diseases in Australia: Their prevalence, consequences and prevention. Australian Institute of Health and Welfare
2. Asthma Foundation of New South Wales: Federal budget - a missed opportunity to ease burden of asthma
3. Berleant, D.: Medical Informatics: Knowledge Management and Data Mining in Biomedicine. Springer Science and Business Media (2005)
4. Fayyad, U., Piatetsky-Shapiro, G.S.P.: Knowledge discovery and data mining: Towards a unifying framework. In: Proceedings of the 2nd ACM international conference on knowledge discovery and data mining (KDD), Portland, OR, USA, pp. 82–88 (1996)
5. Kudyba, S.: Managing data mining: Advice from experts (2004)
6. Kriegel, H.-P., Borgwardt, K.M., Kröger, P., Pryakhin, A., Schubert, M., Zimek, A.: Future trends in data mining. *Data Mining and Knowledge Discovery* 15, 85–97 (2007)
7. Krzysztof, J., Cios, G.W.M.: Uniqueness of medical data mining. *Artificial Intelligence in Medicine* 26, 1–24 (2002)
8. Peter, V., Mike, S., John, C., Maree, B., Sharon, B., Colin, R.: Asthma among school children in the barwon region of victoria. *Medical Journal of Australia* 187(4), 221–224 (2007)
9. Shearer, C.: The CRISP-DM model: The new blueprint for data mining. *Journal of Data Warehousing* 5(4) (2000)
10. Burn, A.: Designing effective statistical graphs. *Handbook of Statistics* 9 (1993)
11. Kohonen, T.: *Self-Organizing Maps*, 3rd edn. Springer, Heidelberg (2001)
12. Agrawal Rakesh, R.S.: Fast algorithms for mining association rules. In: Proceedings of the 20th VLDB Conference, Santiago, Chile, pp. 487–499 (1994)

Mining Context-Specific Web Knowledge: An Experimental Dictionary-Based Approach

Vincenzo Di Lecce^{*}, Marco Calabrese, and Domenico Soldo

Polytechnic of Bari – II Faculty of Engineering - DIASS

Polytechnic of Bari

{v.dilecce,m.calabrese,d.soldo}@aeflab.net

Abstract. This work presents an experimental semantic approach for mining knowledge from the World Wide Web (WWW). The main goal is to build a context-specific knowledge base from web documents. The basic idea is to use a reference knowledge provided by a dictionary as the indexing structure of domain-specific computed knowledge instances organised in the form of inter-linked text words. The WordNet lexical database has been used as reference knowledge for the English web documents. Both the reference and the computed knowledge are actually conceived as word graphs. Graph is considered here as a powerful way to represent structured knowledge. This assumption has many consequences on the way knowledge can be explored and similar knowledge patterns can be identified. In order to identify context-specific elements in knowledge graphs, the novel semantic concept of “minutia” has been introduced. A preliminary evaluation of the efficacy of the proposed approach has been carried out. A fair comparison strategy with other non-semantic competing approaches is currently under investigation.

Keywords: Semantic Web, Web Mining, Knowledge Discovery, WordNet.

1 Introduction

Tim Berners-Lee introduced the term “Semantic Web” in 2001 with the definition [1]: “The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation”. The Semantic Web approach has the purpose of providing machines with the ability of understanding the semantic content of a website. In the W3C intentions, the Semantic Web is the place where agents operate with the tasks of: understanding the semantic content of web pages; creating efficient routes in response to the given queries; replacing the user; creating connections among the sites and verifying the result plausibility.

Several problems however arise when dealing with semantics in web documents. In fact, it can be easily verified, that there can be found two information sources that are semantically similar but morphologically disparate due to the lack of a standard reference for the creation of any kind of document. The proposed system endeavours

^{*} Corresponding author.

to overcome this problem by generating a correspondence between morphological structure and semantic structure. The link structure of a website is related to the corresponding word graph taken from its structure. This graph structure is then referred, by an appropriate pseudo-metric, to the word graph provided by a dictionary used as reference knowledge. Up to the authors' knowledge the idea of taking semantics from web link texts is quite novel in the literature. Nevertheless, some underpinning concepts such as the use of a reference and a computed knowledge can be found in other authors as the related works section explains.

The outline of the paper is as follows. Section 2 sketches some well-known methods particularly used to discover knowledge from web pages. Section 3 presents the proposed system and the following one presents results obtained from experiments. Conclusions are given in section 5.

2 Related Works

For the extraction of Web sub-graphs, it is interesting to mention the method presented by Kleinberg [2]. In [2] the author proposes a method to extract relevant sources from document set being not completely interconnected, in response to a given query. Particularly relevant in the WWW domain, the proposed system is a link-based model for assigning authoritative pages following general research topics. The model is based on the relationship that exists between the authorities for a topic and those pages – hubs – that link to many related authorities. Kleinberg discovers the existence of a natural balance between hubs and authorities in the graph defined for a hypertext; this leads to the implementation of an algorithm for the identification of both above-quoted categories. The algorithm is applied to the focused WWW sub-graphs that are produced as output of a common search engine. The algorithm provides small sized sets with documents that are relevant to a given topic.

The FCA (Formal Concept Analysis) is a method mostly used for the data analysis (e.g., the extraction of the implicit relation among the objects described through a set of attributes). This can be considered as a conceptual clustering technique. In [3] the authors present a method based on the FCA for the automatic acquisition of taxonomy or hierarchical concept from a textual document. First of all the text is labelled through POS (Part-Of-Speech) and then analysed to obtain a syntactic tree for every phrase. The verb/subject, verb/object and verb/prepositional phrase dependencies are extracted from the syntactic trees. These collection couples are filtered to delete low-level data and then they are weighted with statistical methods; all the couples exceeding the given threshold of weight level are converted in a formal context to which the FCA analysis is applied. The graph obtained from the FCA analysis is turned into another one which contains all the objects nearest to a conceptual hierarchy.

In [4] Seo et Al. analyse the main statistical methods used to extract ontology relevant concept sets. The evaluation is realized by comparing the ontology obtained from each feature selection method with domain ontology manually assigned. The considered methods for the feature extraction in a dataset are: Mutual Information, χ^2 Statistic, Markov Blanket and Information Gain. In [4] the authors show the Mutual Information and χ^2 Statistic methods are better than the others; these methods are used in information retrieval for their ability to identify the words with higher semantic content for a class, respecting the ontology definition.

In [5] a standard based evaluation of ontology learning through a lexical term analysis is proposed. The work highlights the difference between two term layers: lexical layer and hierarchical concept layer. The first layer is linked to the sets of all ontology related terms, while the second layer depends on the semantic structure of the same ontology. The evaluation of the considered ontology (computed retrieval) is measured in relation to a reference ontology (reference retrieval). Among the testing learning techniques, the authors present an approach that compares the ontology analysis result with a given standard. This solution is especially suitable for broad scale evaluations and in learning methods of more ontologies.

The afore-mentioned works about the creation of taxonomies and ontologies emphasize an important aspect: the learning of conceptual hierarchies cannot be obtained in completely automatic way; in this area human role is necessary to assign the dataset context and the ontology context. Although the current work does not specifically analyse ontologies it introduces a tool to extract semantically relevant information contained in one or more websites by means of contextual similarity evaluator for the web documents. This method is consistent with what stated in [5] but differently, in this work, its application is considered at a lower abstraction layer, to be more precise, at the taxonomy layer (taxonomy envisages the concept classification – hierarchy – while ontology allows for the formulation of complete conceptual pattern in a given website).

3 Proposed System Overview

The adopted web mining technique requires two distinct phases: supervised pre-processing and unsupervised post-processing.

- (Phase I – crawling/parsing): During the first phase some websites are classified into an a-priori chosen context. This step consists in the user manually providing relevant starting URIs to the system crawlers. Alternatively, the system may be also fed by URIs collected from the query result of a search engine (these URIs are generally ordered by a decreasing relevance index – e.g., the Page Rank weight calculated with the Page Rank Algorithm [6]). Nevertheless, the system requires the human role as the input process for primigenial website categorisation.
- (Phase II - semantics) The post-processing step refers to the automatic extraction of semantics from the document set. The system implements a novel technique to assess context-relevant concept words using the two concepts of reference knowledge ([7] i.e. ground-truth knowledge base) and computed knowledge (i.e. extracted knowledge). WordNet [8], a lexical database for the English language provided by the Princeton University, is used here as reference knowledge. The underpinning assumption is that knowledge can be structured as a graph where concept words (graphically represented by vertices) refer to other concept words. Thus, the concept patterns among adjacent vertices become assertions in a given context domain. The distances among given concept patterns can be evaluated once a proper distance measure among vertices has been defined.

3.1 Structuring Knowledge: The Key Concept of “Minutia”

Let $G_r = (V_r, E_r)$ and $G_c = (V_c, E_c)$ be two graphs corresponding respectively to the reference and the computed knowledge (actually, as will be pointed later, there may be as many G_c graphs as there are different non-interlinked web sites to analyse). Each vertex in a graph is labelled with a single word or a sequence of words (in this last case we speak as of locution). The labelling process will be explained further in the text. Let G be the union of the two graphs $G_c \cup G_c$ assuming that vertices with the same label are considered only once, thus representing connection points for the two graphs. By taking a couple of adjacent vertices on G_c (i.e. V_{c_x}, V_{c_y} with V_{c_x} pointing to V_{c_y}) it is generally possible to find two paths that, moving from both vertices, intersect each other on a G_r vertex. We call this G_r vertex *minutia* (Figure 1).

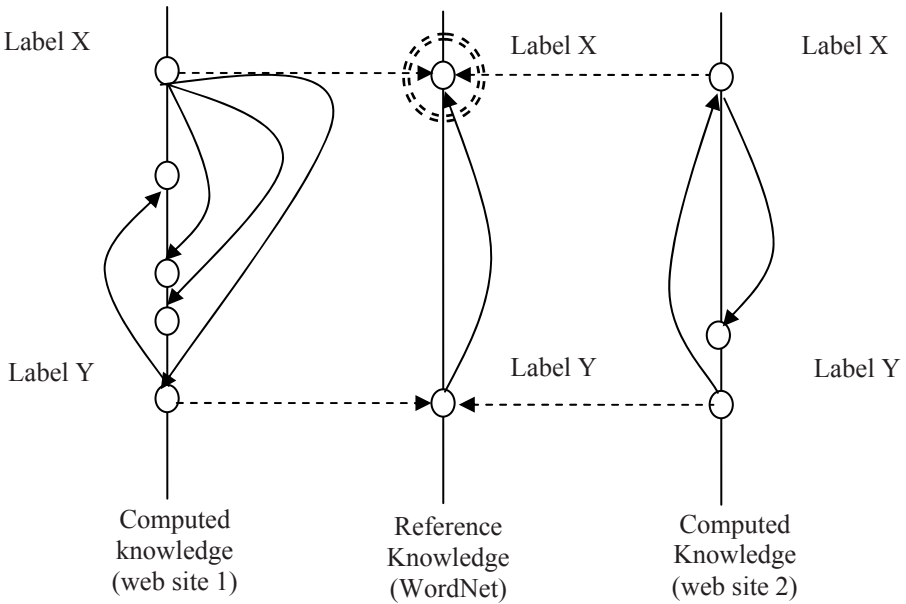


Fig. 1. 1-order minutia example for two different web sites corresponding to two different computed knowledge threads. Label X vertex points to Label Y vertex on the computed knowledge thread 1 and Label Y points to Label X on the computed knowledge thread 2. Label X on the reference knowledge thread is found as the connection point of the two paths resulting from Label X and label Y on the computed knowledge threads.

The *order* of a minutia is defined as the number of edges on G characterizing the longest one of the two paths connecting the two adjacent vertices with the minutia itself. 0-order minutiae correspond to couples of adjacent vertices in G_r that are labelled alike. In this work only 0-order and 1-order minutiae have been considered. Figure 1 depicts a 1-order minutia example where two different web sites intercept the same minutia, although they do not have the same morphology.

The same considerations hold true if G_c represents a web site structure where vertices are weblinks, edges are webpages and labels are the weblink texts. Extending the

procedure of finding minutiae to all the couples of adjacent links that can be found in a given web site is an easy task. The outcome of this search is a list of minutiae ranked by their occurrences. By definition, every minutia corresponds to one or more couples of words (locutions) in the computed knowledge base, hence a couple of words (locutions) extracted by two weblink texts. This means that there is a 1:N correspondence between minutiae and web pages, i.e. a minutia may pertain to a list of web pages. Computing the distance between two given minutiae corresponds to evaluating the mutual distances between the corresponding sets of link words.

3.2 System Architecture

With the purpose of assessing the semantic relevance of the concept of minutia, a web-based software tool has been implemented. The system is mainly composed of two logical components. These are:

- Knowledge Manager (KM): that handles web crawling and parsing activities and the analysis of the website structure.
- Web Evaluator (WE): that manages the used dictionaries, finds minutiae and performs semantic-contextual similarity evaluation between websites.

3.2.1 Knowledge Manager

This component manages the operations of gaining information from web sources and storing it in a MySQL database that can be split into three conceptually semi-independent parts:

1. Page repository: it represents the core of the system which stores interlinked web pages along with their graph structure (a pointer-to-pointer table is used for this scope, that is a table that memorises the couples of adjacent links);
2. Synonym dictionary: it represents the system reference knowledge stored in a graph structure similar to Page Repository where each word (or locution) points to its synonyms;
3. Context synonym dictionary: the document semantic content is contextualised for providing a semantic-contextual dictionary (also memorised in a pointer-to-pointer fashion).

The KM performs web structure mining, web content mining and their integration [9]. Web mining activity [10] is generally focused on the structure and content analysis. These two issues are considered closely related in literature and are managed by a web crawler and a parser respectively. A spider has been realised to explore the structure of a document set, whose architecture is obtained by a crawler and a page repository. Unlike other common tools, KM has the peculiarity of working also on a context-dependent dictionary. The context synonym dictionary is built up as the result obtained by extraction of contextual information memorised in the link textual descriptions found in web pages (Figure 2). From this assumption a table of words (locutions) pointing to other words (locutions) is produced. This table of pointers represents the context synonym dictionary which describes the explored website. The resulting computed dictionary (table of pointers) can be opportunely merged with the reference synonym dictionary by means of the pseudo-metric introduced by the concept of minutia.

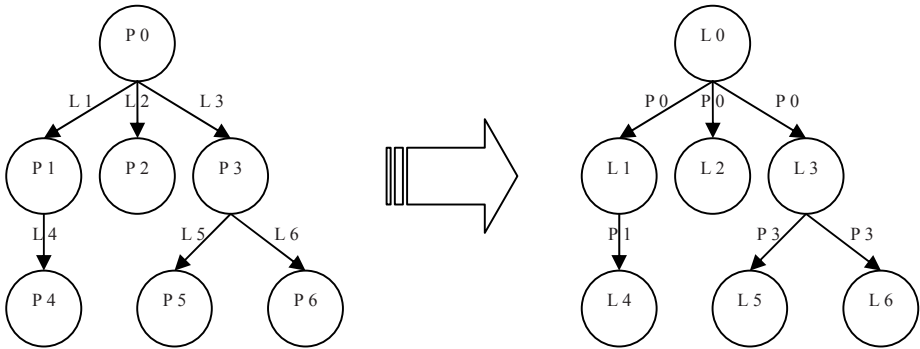


Fig. 2. Two representations of a web-graph. (1) link graph (left side): vertices are web documents and edges are the hyperlinks; P0 represents the home page; (2) word graph (right side): vertices are hyperlink texts and edges are the web documents. L0 is added as a fake link pointing to the home page.

3.2.2 Web Evaluator

The WE module performs contextual and semantic content analysis by querying information to the KM module. A first activity performed by WE concerns the population of the context synonym dictionary. It requires two inputs: context name and the URI representing the entry in the table of pointers.

The authors assume that, referring to a web document, the semantic layer can be derived from the morphological layer – i.e. the link structure. This assumption has been positively tested by numerous experiments as it will be further presented. The graphs obtained from the two layers are similar, as shown in Figure 2. In the link graph the vertex (P_i) and the oriented edge (L_i) represent a page and a hyperlink respectively; while in the word graph the vertex (L_i) and the oriented edge (P_i) refer to a hyperlink and a page respectively. The word graph is extracted by analysing the hyperlink text – here also called locution; thus a page becomes a bridge among locutions. These elements and their semantics are depicted in Table 1.

Another task performed by the WE module concerns the evaluation of the distance between the vertices of a web-graph. The Dijkstra algorithm has been chosen. This algorithm is widely used in telecommunications for data packet routing. Therefore, this algorithm can be applied to the web graph without difficulty. The relationships of semantic synonyms existing between the context-specific terms, by means of the Dijkstra algorithm, are analysed thus obtaining proximity matrix as output. This square matrix upper triangular is of order n , where n is the number of terms present in the dictionary. It is necessary to define a threshold value for an acceptable processing time. The threshold value defines a constant maximum distance. Finally the WE module evaluates the semantic-contextual similarity among websites. This analysis is the result obtained by integrating the WE activities: evaluation of the contextual similarity among the pages and the creation of the context synonym dictionary. It is then possible to define the feature terms (minutiae) extracted by the word graph by evaluating the synonymy among the locutions. If locutions are made of more than one word such a relationship is searched for every tokenised term. These feature terms represent the feature words for the website. The feature words for more websites are

Table 1. Link graph and word graph semantics in comparison

Graph element	Link Graph semantics	Word Graph semantics
P_i	Relation among web links (pointers)	Relation among locutions (synonymy)
L_i	web links (information sources)	Locutions (dictionary entry points)

obtained by extending this notion to a set of websites. Finally the feature words for a context are obtained by considering only common feature words for the chosen set of websites (i.e using the “AND” Boolean conjunction).

4 Experiments and Results

It is hard to find a homogeneous workbench where to compare the proposed system with other non-semantic ones. The paradigm between the two approaches is rather different since they also have different objectives and ways of use. While in traditional search engine techniques the core element is the term-document data structure, in this experimental architecture the leading indexing structure is the one offered by a lexical database like WordNet. A non-semantic search engine provides web documents in response to user query, while a semantic search engine should retrieve concepts refining user query linked to some documents. The first step should be finding an evaluation strategy suitable for a fair comparison between the two approaches. This is currently under investigation. In the meanwhile, an alternative strategy for assessing the quality of the proposed technique has been used. We have made our crawler run on a set of semantically similar web domains: universities, low-cost airlines, seaports. For each domain the set of more frequent minutiae has been considered. Then, a simple information retrieval index has been computed according to the following formula:

$$p_i^\delta = \frac{\#\mu_i^\delta}{\#\lambda_i^\delta} \quad (1)$$

where δ represents the chosen context domain (i.e. “university”), i represents the explored web domain (e.g. www.manchester.ac.uk), $\#\lambda_i^\delta$ is the number of locutions extracted in the whole web domain and $\#\mu_i^\delta$ is the number of minutiae which were found. The index in (1) should give an estimate of the frequency of context relevant concept words (minutiae) in relation to the whole number of found locutions. In the case of the university domain, for example, three minutiae have been found in every web site: these are the tokens “university”, “body” and “educational institution” which intuitively have high semantic closeness. Figure 3 represents the scatter plot of the function p_i^δ and the related regression line for each considered context domain. An interesting empirical behaviour is that index in (1) monotonically increases with the number of locutions. The variance of scattered data gives an important semantic indication about the closeness of the feature terms within the web domain in relation to the general terms characterising the context. Some low cost companies, for example, adopt a marketing strategy that ranges from the core cheap-flight service to other

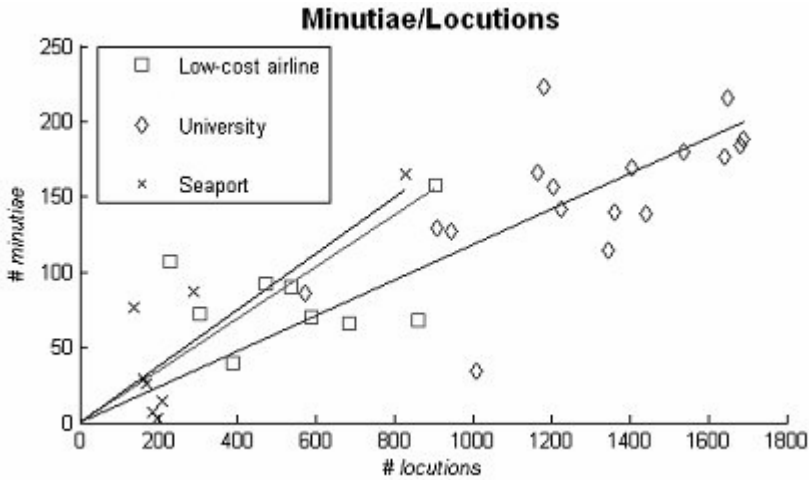


Fig. 3. Scatter plot with regression line of three explored context domains (university, low-cost airline and seaport)

Table 2. Results of the proposed technique for the “university” context domain. The first four columns account respectively for the explored web domain, for the number of distinct minutiae, for the number of locutions (hyperlink texts) and for the retrieval index defined in (1). The fifth column presents an ordered list of found minutiae. They are ranked on their percentage occurrence with respect to the whole number of minutiae (first parameter in parenthesis); second parameter represents the number of web domains where minutiae have been found.

explored web domain	# minutiae	# locutions	p_i^δ	context-specific knowledge Ranked list of minutiae
Manchester	184	1681	0,1095	university (24.95%, 17/17)
Birmingham	34	1008	0,0337	body (16.18%, 17/17)
Southampton	177	1641	0,1078	educational institution (6.81%, 15/17), research (4.57%, 14/17)
Oxford	180	1539	0,1170	alphabetic character (3.51%, 12/17), letter (3.51%, 12/17)
Liverpool	169	1405	0,1203	letter of the alphabet (3.51%, 12/17)
Cambridge	140	1362	0,1028	activity (3.36%, 17/17)
Warwick	114	1346	0,0847	city (2.93%, 12/17)
Brighton	129	908	0,1421	information (2.11%, 17/17)
Edinburgh	166	1164	0,1426	students (2.07%, 15/17)
Nottingham	139	1443	0,0963	content (1.97%, 17/17)
Lancaster	188	1691	0,1112	establishment (1.92%, 14/17), international (1.68%, 15/17)
Kingston	216	1648	0,1311	education (1.56%, 14/17)
Loughborough	142	1224	0,1160	assemblage (1.36%, 16/17)
Aston	223	1180	0,1890	business (1.33%, 14/17)
Brunel	127	944	0,1345	accumulation (1.15%, 13/17)
Northumbria	86	572	0,1503	collection (1.12%, 13/17)
Bournemouth	157	1207	0,1301	study (1.09%, 15/17)
				aggregation (1.09%, 13/17)
				learning (1.04%, 14/17)
				...

surrounding services such as hotel booking or car rental services. In this sense, the system mines the diversity of marketing approaches. It is not a case if, in the university context such discrepancies among web domain are less significant. Table 2 represents the results of the technique here proposed for the university domain.

5 Conclusions

In this work, an experimental method for mining semantic-contextual similarity among websites has been presented. This method is based on the application of a semantic similarity evaluator among locutions derived from a web-graph structure. Both reference and computed knowledge have been structured as graph of words pointing to other words. The basic hypothesis is that, referring to a web document, the semantic layer can be derived from the morphological layer (link structure) by extracting the word graph corresponding to the structure of the web link locutions. The key concept of *minutia* has been introduced with the aim of finding feature terms characterising the given context. This hypothesis seems to be confirmed by multiple experiments carried out over different contexts. Many research investigations regarding these concepts are currently under way.

Our future effort will be devoted to the following topics:

1. evolution of reference knowledge by means of artificial intelligence systems;
2. creation of post-processing refinement morphological methods;
3. realisation of a finished version of a semantic search engine based on the proposed techniques;
4. extraction of ontologies starting from the context synonym dictionary. The ontology is in fact a data model that represents a concept set and the existing relationship among these concepts (“An ontology is a formal, explicit specification of a shared conceptualisation” [Gruber, 1993]).
5. finding a way to quantify the evidence that the proposed system largely outperforms non-semantic systems or other competing approaches.
6. exploitation of partially unsupervised intelligent crawling and parsing techniques.

References

1. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. *Scientific American* (May 2001)
2. Kleinberg, J.M.: Authoritative Sources in a Hyperlinked Environment. In: Proc. of the 9th ACM-SIAM Symposium on Discrete Algorithms (SODA 1998), San Francisco, California, USA, January 1998, pp. 668–677 (1998); *Journal of the ACM (JACM)* 46, 604–632 (September, 1999) (Extended version)
3. Cimiano, P., Hotho, A., Staab, S.: Learning Concept Hierarchies from Text Corpora using Formal Concept Analysis. *Journal of Artificial Intelligence Research (JAIR)* 24, 305–339 (2005)

4. Seo, Y.W., Ankolekar, A., Sycara, K.: Feature Selection for Extracting Semantically Rich Words. Technical report CMU-RI-TR-04-18 Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania (March 2004)
5. Dellschaft, K., Staab, S.: On How to Perform a Gold Standard Based Evaluation of Ontology Learning. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 228–241. Springer, Heidelberg (2006)
6. Brin, S., Page, L.: The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Computer Networks and ISDN Systems* 30, 107–117 (1998)
7. Richardson, R., Smeaton, A.F., Murphy, J.: Using WordNet for Conceptual Distance Measurement. In: Proc. of the Annual BCS-IRSG Colloquium on IR Research, Glasgow, Scotland, pp. 100–123 (March 1994)
8. Fellbaum, C.: WordNet: An Electronic Lexical Database (May 1998) ISBN-10: 0-262-06197-X
9. Chakrabarti, S., Dom, B.E., Gibson, D., Kleinberg, J.M., Kumar, R., Raghavan, P., Rajagopalan, S., Tomkins, A.: Mining the Link Structure of the World Wide Web. *IEEE Computer* 32, 60–67 (1999)
10. Kosala, R., Blockeel, H.: Web Mining Research: A Survey. *ACM SIGKDD Explorations Newsletter* 2, 1–15 (2000)

Mining Direct and Indirect Fuzzy Multiple Level Sequential Patterns in Large Transaction Databases

Weimin Ouyang^{1,2}, Qinhuang Huang¹, and Shuanghu Luo²

¹ Modern Education Technology Center, Shanghai University of Political Science and Law
200438 Shanghai, China

{oywm, qinhuahuang}@shupl.edu.cn

² School of Computer Engineering and Science, Shanghai University,
200072 Shanghai, China

shluo@staff.shu.edu.cn

Abstract. Sequential pattern is an important research topic in data mining and knowledge discovery. Traditional algorithms for mining sequential patterns are built on the binary attributes databases, which has three limitations. The first, it can not concern quantitative attributes; the second, only direct sequential patterns are discovered; the third, it can not process these data items with multiple level concepts. Mining fuzzy sequential patterns has been proposed to address the first limitation. We put forward a discovery algorithm for mining indirect multiple level sequential patterns to deal with the second and the third limitations, and a discovery algorithm for mining both direct and indirect fuzzy multiple level sequential patterns by combining these three approaches.

1 Introduction

Just like association rule [1], Sequential pattern [2][3] is an important research topic in data mining and knowledge discovery, which is firstly proposed by R.Agrawal. While association rules mining is to find the intra-transaction relations, sequential patterns mining is to find the inter-transactions relations. A sequential pattern is formed as (A, B), where A and B are disjoint item sets, and its support is no less than a user-specified minimum support. The sequential patten (A, B) means that if A is in a transaction, then B would be in another transaction with high probability.

Traditional algorithms for mining sequential patterns are built on the binary attributes databases, which has three limitations. First, it can not concern quantitative attributes; second, it can not process these data items with multiple level concepts; third, only direct sequential patterns are discovered.

In the real applications, not only may have binary attributes, but also may have quantitative attributes. T. P. Hong put forward a method of mining fuzzy sequential patterns from quantitative transaction database to address the first limitation^[4], which first transforms quantitative values in transactions into linguistic terms, then filters them to find sequential patterns by modifying the AprioriAll mining algorithm.

During the discovery of sequential patterns, only the frequent itemsets are considered while all the infrequent itemsets are ignored. Recently, researchers have recognized that some infrequent patterns can provide very useful insight view into the data

set, and a new kind of knowledge discovery problems called as indirect associations has been proposed [5]. Consider a pair of item x and y , which are rarely present together in the same transaction. If both items are highly dependent on the presence of another itemsets M , then the pair of x and y is said to be indirectly associated by M called as mediator. Inspired by this idea and multiple level association rules, we put forward indirect sequential pattern and its discovery algorithm in this paper to deal with the second limitation.

In this paper, we propose an approach for mining both direct and indirect fuzzy multiple level sequential patterns by integrating indirect, fuzzy and multiple level concept extensions to sequential patterns.

The rest of this paper is organized as follows. The definitions for fuzzy direct and indirect sequential pattern are given in Section 2. In Section 3, we describe the discovery algorithm for mining fuzzy direct and indirect sequential patterns. Section 4 presents our primary experimental results. The conclusions are made in the last section.

2 Problem Definition

2.1 Sequential Patterns

Let $I = \{i_1, i_2, \dots, i_m\}$ be a set of literals called items. Let the database $D = \{t_1, t_2, \dots, t_n\}$ be a set of n transactions, where each transaction is a subset of I . A non-empty subset of I is called itemset. An itemset containing k items is called k -itemset. The support of an itemset X denoted as $\text{sup}(X)$ is defines as the fraction of all transactions containing X in D . An itemset is frequent if its support is greater than a user-specified threshold minisup .

A sequence is an ordered list of itemsets such as $s = (s_1, s_2, \dots, s_u)$, where each itemset s_i is an element of the sequence.

A sequence $t = (t_1, t_2, \dots, t_v)$ is called a subsequence of s if there exist integer $1 \leq j_1 < j_2 < \dots < j_v \leq u$ such that $t_1 \subseteq s_{j_1}, t_2 \subseteq s_{j_2}, \dots, t_v \subseteq s_{j_v}$. A sequence is called sequential pattern if the support of the sequence is no less than the predefined minimum support threshold minisup .

2.2 Direct and Indirect Sequential Patterns

We put forward the concept of indirect sequential patterns. As a result, traditional sequential pattern is called as direct sequential pattern.

Let sequence s_1 be (a_1, a_2, \dots, a_k) , and sequence s_2 be (b_1, b_2, \dots, b_k) . The indirect sequential pattern is defined as follows:

Definition 1: The pair of sequences s_1 and s_2 is said to joinable if one of the following conditions is hold:

- (1) $a_i = b_i$ for $i = 1, 2, \dots, k-1$;
- (2) $a_i = b_i$ for $i = 2, 3, \dots, k$;
- (3) $a_i = b_{i-1}$ for $i = 2, 3, \dots, k$;

In the context of indirect sequential patterns, we set $a_k = x, b_k = y, (a_1, a_2, \dots, a_{k-1}) = M$ if condition (1) is satisfied, or set $s_1 = Mx, s_2 = My$; set $a_1 = x, b_1 = y, (a_2, a_3, \dots, a_k) = M$

if condition (2) is satisfied, or set $s_1 = xM, s_2 = yM$; set $a_1 = x, b_k = y, (a_1, a_2, \dots, a_{k-1}) = M$ if condition (3) is satisfied, and $s_1 = xM, s_2 = My$.

Definition 2: A pair of itemsets x and y is said to be a indirect sequential pattern via a mediator sequence M , if $s_1 = Mx$ (or $s_1 = xM$), $s_2 = My$ (or $s_2 = yM$) and the following conditions are satisfied:

- (1) $\text{sup}(\{x, y\}) < t_s$;
- (2) There exists a non-empty set M such that:
 - (a) $\text{sup}(s_1) \geq t_f, \text{sup}(s_2) \geq t_f$,
 - (b) $\text{dep}(\{x\}, M) \geq t_d, \text{dep}(\{y\}, M) \geq t_d$, where $\text{dep}(A, B)$ is a measure of the dependence between itemsets A and B .

The thresholds t_s, t_f and t_d are called sequencepair support threshold, mediator support threshold, and dependence threshold, respectively. We usually set $t_f \geq t_s$ in practice.

2.3 Fuzzy Sequential Patterns

In the context of fuzzy sequential patterns, the support of an itemset can be counted as follows: for every sequence in the customer sequences database, take the fuzzy logic AND of the membership values of the items under consideration, and summate these numbers. Let the customer sequences database be SD and an itemset $X = \{x_1, x_2, x_3, \dots, x_k\} \subseteq I$. The support of a sequence $s \in SD$ to the itemset X can be defined as

$$\text{sup}(X, s) = \bigcap_{i=1}^k \mu_{x_i}(s) \tag{1}$$

If we take the fuzzy logic AND as the product, the support of X from the customer sequences database SD is then defined as

$$\text{sup}(X) = \sum_{s \in SD} \text{sup}(X, s) = \sum_{s \in SD} \prod_{i=1}^k \mu_{x_i}(s) \tag{2}$$

If we take the fuzzy logic AND as the min function, the support of X from the customer sequences database SD is then defined as

$$\text{sup}(X) = \sum_{s \in SD} \text{sup}(X, s) = \sum_{s \in SD} \min_{i=1}^k \mu_{x_i}(s) \tag{3}$$

2.4 Multiple Levels Sequential Patterns

Most algorithms for mining sequential patterns focused on finding sequential patterns on the single level concept. However, data item taxonomies are usually predefined in the real-world applications, which can be represented as concept hierarchy trees such as Fig 1. Terminal nodes on the trees represent actual items appearing in transactions; internal nodes represent classes or concepts formed by lower-level nodes.

In Fig. 1, the root node is at level 0, the internal nodes representing categories (such as “milk”) are at level 1, the internal nodes representing flavors (such as “plain”) are at level 2, and the terminal nodes representing brands (such as “A”, “B”, “C”) are at level 3. Only terminal nodes appear in transactions. Nodes in predefined

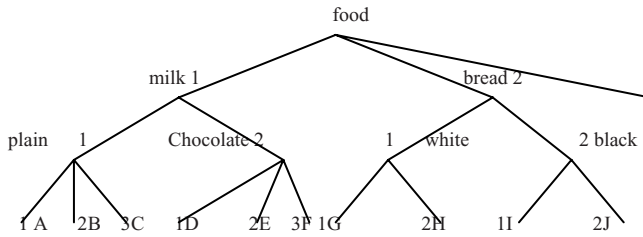


Fig. 1. concept hierarchy trees

taxonomies are first encoded using sequences of numbers and the symbol “*” according to their positions in the hierarchy tree. For example, the internal node “Milk” in Fig. 1 would be represented by “1**”, the internal node “plain” by “11*”, and the terminal node “A” by “111”, where “*” is an asterisk wildcard.

3 Discovery Algorithm for Mining Direct and Indirect Fuzzy Multiple Level Sequential Patterns

We put forward the idea of indirect sequential patterns to resolve the second limitation, and a discovery algorithm for mining both direct and indirect fuzzy sequential patterns. Notations used in our algorithm are described as follows.

n : The total number of transactions in database;

x : The number of levels in a given taxonomy;

$m[L]$: The total number of items at level L , $1 \leq L \leq x$

c : The total number of customers

n_i : The number of transactions from the i th customer, $1 \leq i \leq c$

D_i : The purchase sequence of the i th customer, $1 \leq i \leq c$

D_{id} : The d th transaction in D_i , $1 \leq d \leq n_i$

$I^g[L]$: The g th item at level L , $1 \leq g \leq m[L]$

$R^{gk}[L]$: The k th region of $I^g[L]$, $1 \leq k \leq |I^g[L]|$, where $|I^g[L]|$ is the number of fuzzy regions for item $I^g[L]$

$v_{id}^g[L]$: The quantitative value of item $I^g[L]$ in D_{id}

$f_{id}^g[L]$: The membership value of $v_{id}^g[L]$ in region $R^{gk}[L]$

$f_i^{gk}[L]$: The membership value of region $R^{gk}[L]$ in the i th customer sequence D_i

$\text{count}^{gk}[L]$: The scalar cardinality of region $R^{gk}[L]$

$\text{maxcount}^g[L]$: The maximum count value among $\text{count}^{gk}[L]$ values

$\text{maxR}^g[L]$: The fuzzy region of item $I^g[L]$ with $\text{maxcount}^g[L]$

$C_k[L]$: the set of candidate itemsets with k items at level L

$L_k[L]$: the set of large itemsets with k items at level L

$SC_k[L]$: the set of candidate sequences with k itemsets at level L

$SL_k[L]$: the set of large sequences with k itemsets at level L

$SNL_k[L]$: The set of non-large sequence with k itemsets at level L

minsup[L]: The predefined minimum support value at level L
 t_s[L]: The predefined sequencepair support threshold at level L;
 t_f[L]: The predefined mediator support threshold at level L
 t_d[L]: The predefined dependence threshold at level L
 The proposed mining algorithm is described as follows.

Algorithm: MDIFMSP(Mining both direct and indirect fuzzy multiple level sequential patterns)

Input: A body of n transaction data, each consists of customer ID, transaction time and the purchased items with their quantities, a set of membership functions and a predefine minimum support threshold minsup, sequencepair support t_s, mediator support threshold t_f, and dependence threshold t_d;

Output: A set of direct fuzzy sequential patterns DFSP, a set of indirect fuzzy sequential patterns IFSP;

- (1) Encode the predefined taxonomy using a sequence of number and the symbol “*” according to their positions in the hierarchy tree.
- (2) Translate the item names in the transaction data according to the encoding scheme.
- (3) The transaction database is sorted by customer ID as the major key and transaction time as the minor key.
- (4) Set L=0, L is used to store the level number to be processed.
- (5) Set L=L+1.
- (6) Form a customer purchase sequence D_i for each customer i by sequentially listing his/her n_i itemsets, where n_i is the number of transactions from customer i. Denote the dth itemset in D_i as D_{id}.

- (7) Transform the quantitative value v^g_{id}[L] of each itemset I^g[L] appearing in D_{id} into a fuzzy set f^g_{id}[L] represented as

$$\left(\frac{f_{id}^{g1}[L]}{R^{g1}[L]} + \frac{f_{id}^{g2}[L]}{R^{g2}[L]} + \dots + \frac{f_{id}^{gl}[L]}{R^{gl}[L]} \right)$$

using given membership functions,

where R^{gk}[L] is the kth fuzzy region of item I^g[L], f^{gk}_{id}[L] is v^g_{id}[L]’s fuzzy membership value in region R^{gk}[L], and l is the number of fuzzy regions for I^g[L].

- (8) Find fuzzy membership value f^{gk}_i[L] of each region R^{gk}[L] in each customer sequence Di as f^{gk}_i[L] = MAX^{|D_i|}_{d=1}(f^{gk}_{id}[L]), where |D_i| is the number of itemsets in D_i.
- (9) Caculate the scalar cardinality of each attribute region D^{gk}[L] as count^{gk}[L] = ∑_{i=1}^c f^{gk}_i[L].

- (10) Find maxcount^g[L]= MAX^l_{k=1}(count^{gk}[L]), where 1 ≤ g ≤ m[L] and l is the number of regions for item I^g[L]. Let maxR^g[L] be the region with maxcount^g[L] for item I^g[L]. The maxR^g[L] will be used to represent the fuzzy characteristic of item I^g in later mining processes.

- (11) Check whether the value $\max\text{count}^g[L]$ for a region $\max R^g[L]$, for $g=1$ to $m[L]$, is larger than or equal to the predefined minimum support threshold $\text{minsup}[L]$. If the value of $\max\text{count}^g[L]$ is equal to or greater than $\text{minsup}[L]$, put $\max R^g[L]$ in the large 1-itemsets $L_1[L]$. That is $L_1[L]=\{ \max R^g[L] \mid \max\text{count}^g[L] \geq \text{minsup}[L], 1 \leq g \leq m[L] \}$.
- (12) If $L_1[L]$ is null, then goto the step (5); otherwise, do the next step.
- (13) Set $k=1$, where k is used to represent the number of items kept in the current large itemsets.
- (14) Generate the candidate set $C_{k+1}[L]$ from $L_k[1], L_k[2], \dots, L_k[L]$ in a way similar to that in the apriori algorithm^[1].
- (15) Do the following substeps for each newly formed $(k+1)$ -itemsets s with items $(s_1, s_2, \dots, s_{k+1})$ in $C_{k+1}[L]$:
 - (a) Calculate the fuzzy value $f_i^s[L]$ for s in each customer sequence D_i as:

$$f_i^s[L] = \text{MAX}_{d=1}^{|D_i|} (f_{id}^s[L]), \text{ and}$$

$$f_{id}^s[L] = f_{id}^{s_1}[L] \wedge f_{id}^{s_2}[L] \wedge \dots \wedge f_{id}^{s_{k+1}}[L].$$
 - (b) Calculate the scalar cardinality of s as $\text{count}^s[L] = \sum_{i=1}^c f_i^s[L]$.
 - (c) If count_s is larger than or equal to the predefined minimum support threshold minsup , put s in $L_{k+1}[L]$.
- (16) If $L_{k+1}[L]$ is null, then do the next step; otherwise, set $k=k+1$ and repeat Step 10 to Step 12.
- (17) Map each larger itemset to a contiguous integer and put it in the large 1-sequence set SL_1 .
- (18) Transform each customer sequence using the integer representation.
- (19) Set $w=1$, where w is used to represent the maximum length of sequences currently kept.
- (20) Generate the candidate set $SC_{w+1}[L]$ from $SL_w[L]$ by joining two large w -sequence from $SL_w[L]$ in a way similar to that in the aprioriAll algorithm^[2].
- (21) Do the following substeps for each newly formed $(w+1)$ -sequence $u = (u_1, u_2, \dots, u_{w+1})$ in $SC_{w+1}[L]$ (each u_i is actually an itemset for the original data):
 - (a) Calculate the fuzzy value $f_i^u[L]$ of u in each transformed customer sequence D_i as $f_i^u[L] = \text{MIN}_{k=1}^{w+1} (f_i^{u_k}[L])$, where itemset u_k must appear after itemset w_{k-1} in D_i . If the sequence u occurs more than once, $f_i^u[L]$ is set by maximum fuzzy value of all the occurrences.
 - (b) Calculate the scalar cardinality of u as $\text{count}^u[L] = \sum_{i=1}^c f_i^u[L]$.
 - (c) If $\text{count}^u[L]$ is larger than or equal to the predefined minimum support threshold minsup , put u in $SL_{w+1}[L]$; otherwise put u in $SNL_{w+1}[L]$.
 - (d) If $w \geq 2$, for each $\langle x, y, M \rangle \in SNL_{w+1}[L]$, if $\text{sup}(\{(x, f_x), (y, f_y)\}) < t_s[L]$, $\text{dep}(\{(x, f_x), (M, f_M)\}) \geq t_d[L]$ and $\text{dep}(\{(y, f_y), (M, f_M)\}) \geq t_d[L]$ are all hold, then put the $\langle x, y, M \rangle$ in IFSP.
- (22) If $SL_{w+1}[L]$ is null, then do next step; otherwise, set $w=w+1$, and repeat step 20 to 22.

- (23) If $L \leq x$, then goto the step (5); otherwise, do next step.
- (24) Transform the maximally large q-sequences, $q \geq 2$, into sequences of original items and put them to DFSP.

4 Experiments

To test the performance of our proposed algorithm, we have done some experiments. The computation environments are Pentium D820, memory of 1G, operating system of Windows XP. The algorithm is implemented with C++. The number of levels was set to 3. The synthetic experiment data set is generated by Assocgen^[2] program of IBM Almaden research center. The meanings of used parameters are as follows:

- D: Number of customers(size of database)
- C: Average number of transactions per Customer
- T: Average number of items per Transaction
- S: Average length of maximal potentially large Sequences
- I: Average size of Items in maximal potentially large sequences
- N_s : Number of maximal potentially large Sequences
- N_i : Number of maximal potentially large Itemsets
- N: Number of items

We set parameters $C=10, T=5, S=4, I=2.5, N_s =500, N_i =2500, N =10000$, total number of customers $D=100000$, and the generated dataset is named as C10T5S4I25.

Figure 2 shows the algorithm executing time variance with minimum support decreasing from 1% to 0.2%. It shows that the algorithm increases with the declining of minsup.

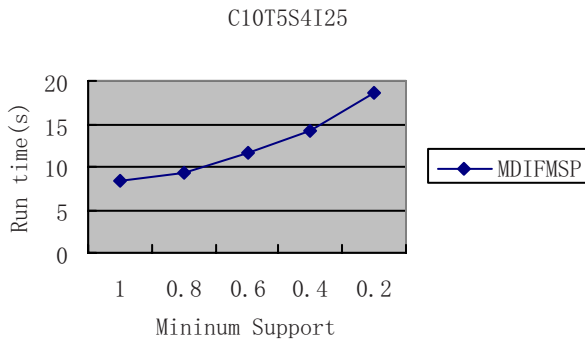


Fig. 2. Execution times

5 Conclusions

In this paper, we addressed three limitations of traditional sequential patterns generalized sequential patterns mining: crisp itemsets, only direct sequential patterns and only single level concept sequential patterns. The fuzzy extension of crisp itemsets

results in approaches of mining fuzzy sequential patterns, indirect sequential patterns can be discovered while direct sequential patterns are mined, and the data item taxonomies extension of data item results in approaches of mining multiple level sequential patterns. We put forward an algorithm for mining both direct and indirect fuzzy multiple level sequential patterns by combining these three extensions. The primary experiments demonstrated that the algorithm is efficient and scalable.

Acknowledgements. We would like to thank T. P. Hong for his kindly email some reference literatures to us and excellent comments on our research works.

References

1. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules. In: The Proc. of the 20th International Conference on VLDB, Santiago, pp. 487–499 (1994)
2. Agrawal, R., Srikant, R.: Mining Sequential Patterns. In: The Proc. 1995 Int. Conf. on Data Engineering, Taibei, Taiwan (March 1995)
3. OuYang, W.M., Cai, Q.S.: Automatic Discovery of Generalized Sequential Pattern in Database. *Journal of Software* 8(11) (November 1997)
4. Hong, T.P., Lin, K.Y., Wang, S.L.: Mining Fuzzy Sequential Patterns from Quantitative Transactions. *Soft Computing* 10(10), 925–932 (2006)
5. Tan, P.N., Kumar, V.: Indirect Association: Mining Higher Order Dependences in Data. In: Proc. Of the 4th European Conference on Principles and Practice of Knowledge Discovery in Databases, Lyon, France, pp. 632–737 (2000)

Rough Set Approaches for Mining Incomplete Information Systems

M.K. Sabu¹ and G. Raju²

¹ MES College, Aluva, Kerala, India
sabu.mes@rediffmail.com

² SCMS School of Technology and Management, Cochin, Kerala, India
kurupgraju@rediffmail.com

Abstract. Rough Set theory and its extensions are found to be very effective in Knowledge Discovery in Data Bases. A number of tools are available in RST to solve Data Mining tasks such as clustering, rule mining, handling missing values and elimination of redundant data. In Data Mining handling of information tables with missing data values plays a very important role as missing values reduces the quality of information extracted. In this paper we discuss three different approaches to handle missing values. First method is called RSFit approach which predicts the missing attribute values based on a distance function. Second method, called Characteristic set based approach, provides decision rules from incomplete information systems. Finally a novel approach is introduced for constructing decision rules. This is based on a similarity relation. Experiment with small data set shows that the new approach is slightly better than the second method.

Keywords: rough sets; indiscernibility relation; missing attribute values; characteristic set; similarity relation; modified similarity relation.

1 Introduction

Rough set is a mathematical approach proposed by Z Pawlak in the early 1980's [1][2][3]. The basic concept of Rough Set theory includes Indiscernibility relation, Lower approximation, Upper approximation, Reduct and Core. Rough sets are applied in many domains such as medicine, finance, image analysis, pattern recognition etc. Rough set theory and its extensions are found to be very effective in data mining [4][5] tasks such as data reduction, handling missing values, feature selection and feature extraction.

Missing attribute values are common in any real world data set. Even a small percentage of missing data can cause serious problems during data analysis stage leading to wrong conclusions and predictions. Data values are missing because of two main reasons – either they are lost or they may exist due to do not care conditions. Missing values can be classified into missing at random (MAR), Missing completely at Random (MCAR) and not ignorable (NI) [14]. In MAR, the process that determines whether or not a cell is missing should not depend on the values in the cell. In MCAR, missing values cannot be predicted any better with the information in the data table.

In NI, the probability that a cell is missing depends on the unobserved value of the missing response.

In this paper, we discuss three Rough Set theory based approaches. In predicting missing attribute values, the RST concepts Reduct and Core are effectively utilized. A Reduct is a subset of condition attributes that are sufficient to represent the whole data set. The intersection of all possible Reducts is Core. In all these approaches, we assume that the missing attribute values exist only in condition attributes. In the first approach the incomplete table is converted into a complete table by supplying appropriate value for the missing attribute. In the second and third method Rough Set approaches are applied to derive rules from incomplete tables.

In section 2 we explain the RSFit approach. The Characteristic Set based approach is discussed in Section 3 and in section 4, we introduces a modified Rough Set approach.

2 RSFit Approach to Assign Missing Values

The input to RSFit approach is a decision table $T=(C, D)$, where C is the set of condition attributes and D is the set of decision attributes. $U = \{U_1, U_2 \dots U_n\}$ is the set of data instances in T . The RSFit approach is based on the (attribute, value) pairs contained in the Reduct/Core of T , because the attributes of Reduct/Core are dependent on each other on certain statistic measure [6]. The similar attribute-value pairs for the data instances containing missing values are identified and the most relevant value is supplied.

Let C_k represent the target attribute, for which the value is missing. Generate the Core of the data set $T=(C, D)$. If the target attribute C_k does not belongs to the core consider a Reduct of T . If C_k does not belong to the Reduct we simply add the attribute C_k into the Reduct. Based on the Reduct/Core, a new decision table $T' = (C', D)$ is constructed, where $C' \subseteq C$, is the set of attributes in the Reduct/Core including C_k . To design a match function to predict the target attribute value, select the data instances from this new decision table T' , by considering all other data instances with missing values. When considering the match cases there are two possibilities existing. Consider all data instances or consider data instances having the same decision attribute value. The first possibility is called “global” and second possibility is called “concept” [6]. In this approach the prediction of the missing attribute value is based on a distance function. To define the distance function, let $U_i = (v_{i1}, v_{i2} \dots v_{ik}, \dots, v_{im}, d_i)$, $1 \leq k \leq n'$, be the attribute value pair containing the missing attribute value v_{ik} for C_k , $1 \leq k \leq m'$. Let U_j be any data instance from U . Now the distance from the target data instance U_i to U_j , $dist(U_i, U_j)$, is defined as:

$$dist(u_i, u_j) = \frac{|v_{i1} - v_{j1}|}{\max V_1 - \min V_1} + \frac{|v_{i2} - v_{j2}|}{\max V_2 - \min V_2} \dots \dots \dots + \frac{|v_{im} - v_{jm}|}{\max V_m - \min V_m} \tag{1}$$

For attributes with missing values, the distance is set as 1, the maximum difference between unknown values. The smallest value of $dist(U_i, U_j)$ is considered and conclude that U_j is the best matched object for U_i and the value of the corresponding attribute value in U_j is assigned to the target attribute C_k in U_i . In case there are multiple match cases, randomly select any one for C_k . For non-numeric attributes, convert them into numeric ones during the preprocessing stage.

We applied this approach in a data set as shown in Table 2, which is obtained from Table 1 by replacing one attribute value with “?”.

Table 1. Sample data set

Case	Temperature	Headache	Nausea	Cough	Flu
1	high	yes	no	yes	yes
2	very_high	yes	yes	no	yes
3	high	no	no	no	no
4	high	yes	yes	yes	yes
5	normal	yes	no	no	no
6	normal	no	yes	yes	no

Table 2. Data set with missing attribute

Case	Temperature	Headache	Nausea	Cough	Flu
1	high	yes	no	yes	yes
2	very_high	yes	yes	no	yes
3	high	no	no	no	no
4	high	?	yes	yes	yes
5	normal	yes	no	no	no
6	normal	no	yes	yes	no

A reduct of this data set is {Temperature, Headache, Nausea} and the target attribute Headache belongs to this reduct. Then a new data set is constructed by considering reduct attributes and the decision attribute. By applying the distance function defined above we have to find a value for the attribute Headache in case 4. For RSFit global, we find that the cases 2 and 6 have the smallest difference from the case 4 and hence cases 2 and 6 are the best match cases. We arbitrarily select any one for filling the target attribute value. If we select Headache value of case 2, which is ‘yes’, to fill the target attribute, we get a correct prediction. But if we use the Headache value of case 6, which is no, to fill the target attribute, we get a wrong prediction. For RSFit concept, we find case 2 is the best match case and assign Headache value of this case, that is yes, to fill the target attribute Headache in case 4 again we get a correct prediction.

3 Characteristic Set Based Approach

Any decision table $T=(C, D)$ defines a function f that maps the direct product of U and C into the set of all values, where U is the set all cases. A decision table with an incompletely specified function f is called incomplete decision table. In an incomplete decision table, the lost values will be denoted by “?” and do not care conditions will be denoted by “*”. Let $\mathbf{a} \in A \subseteq C$, and let v be a value of \mathbf{a} for some case. For incomplete decision tables, for an attribute \mathbf{a} , there exist an instance x such that $f(x, \mathbf{a}) = ?$, then x should not be included in any block of attribute value pairs $[(\mathbf{a}, v)]$ for all values v of attribute \mathbf{a} . For an attribute \mathbf{a} , there exist an instance x such that $f(x, \mathbf{a}) = *$, then x should be included in all blocks $[(\mathbf{a}, v)]$ for every possible value of attribute \mathbf{a} . Based on the above argument, the characteristic set $K_B(x)$ is defined as the intersection of blocks of attribute value pairs $[(\mathbf{a}, v)]$ for all attributes \mathbf{a} from B for which $f(x, \mathbf{a})$ is specified

and $f(x, \mathbf{a}) = v$. The characteristic set $K_B(x)$ may be interpreted as the smallest set of cases that are indistinguishable from x using all attributes form B using a given interpretation of missing attribute values [7]. The characteristic relation $R(B)$ is a relation defined on U for $x, y \in U$ as $(x, y) \in R(B)$ if and only if $y \in K_B(x)$.

For incompletely specified decision tables the lower and upper approximations are defined by considering the characteristic sets. Let X be a concept, B be a subset of the set of all attributes and $R(B)$ be the characteristic relation of the incomplete decision table with characteristic set $K_B(x)$. There are two ways of defining lower and upper approximations. The first definition, called subset B -lower approximation of X , defined as $\underline{B}X = \cup\{K_B(x) \mid x \in U, K_B(x) \subseteq X\}$ and subset B -upper approximation is defined as $\overline{B}X = \cup\{K_B(x) \mid x \in U, K_B(x) \cap X \neq \emptyset\}$.

The second possibility is to modify the subset definition of lower and upper approximations by replacing the Universe U from the subset definition by a concept X [7]. Hence the modified definition, concept B -lower approximation of the concept X , is defined as $\underline{B}X = \cup\{K_B(x) \mid x \in X, K_B(x) \subseteq X\}$ and the concept B upper approximation is defined as $\overline{B}X = \cup\{K_B(x) \mid x \in X, K_B(x) \cap X \neq \emptyset\}$.

The subset B -lower approximation of X is same set as the concept B -lower approximation of X . The concept B -upper approximation of X is a subset of the subset B -upper approximation of X . Based on these lower and upper approximations we derive various association rules. Rules induced from the lower approximation of the concept X certainly describe the concept. So they are called certain rules and the rules induced from the upper approximations are called possible rules. Based on these rules we take necessary decisions by considering the missing values.

An example of an incomplete decision table is presented in Table 3. In Table 3 the lost values are denoted by “?” and do not care conditions are denoted by “*”. The blocks of attribute value pairs consistent with the interpretation of missing attribute values, “lost” and “do not care” conditions are as follows:

- [(Temperature, high)] = {1, 4, 5, 8} [(Temperature, very_high)] = {2, 8}
- [(Temperature, normal)] = {6, 7, 8} [(Headache, yes)] = {2, 4, 6, 8}
- [(Headache, no)] = {3, 7} [(Nausea, no)] = {1, 3, 6, 8}
- [(Nausea, yes)] = {2, 4, 5, 7, 8}

For Table 3 and $B=A$, the values of the characteristic sets $K_B(x)$ are:

- $K_A(1) = \{1, 4, 5, 8\} \cap \{1, 3, 6, 8\} = \{1, 8\}$
- $K_A(2) = \{2, 8\} \cap \{2, 4, 6, 8\} \cap \{2, 4, 5, 7, 8\} = \{2, 8\}$
- $K_A(3) = \{3, 7\} \cap \{1, 3, 6, 8\} = \{3\}$
- $K_A(4) = \{1, 4, 5, 8\} \cap \{2, 4, 6, 8\} \cap \{2, 4, 5, 7, 8\} = \{4, 8\}$
- $K_A(5) = \{1, 4, 5, 8\} \cap \{2, 4, 5, 7, 8\} = \{4, 5, 8\}$
- $K_A(6) = \{6, 7, 8\} \cap \{2, 4, 6, 8\} \cap \{1, 3, 6, 8\} = \{6, 8\}$
- $K_A(7) = \{6, 7, 8\} \cap \{3, 7\} \cap \{2, 4, 5, 7, 8\} = \{7\}$ and $K_A(8) = \{2, 4, 6, 8\}$

Table 3. Incomplete decision table with lost and don't care values

Case	Temperature	Headache	Nausea	Flu
1	high	?	no	yes
2	very_high	yes	yes	yes
3	?	no	no	no
4	high	yes	yes	yes
5	high	?	yes	no
6	normal	yes	no	no
7	normal	no	yes	no
8	*	yes	*	yes

From Table 3, the concept A-lower and A-upper approximations are :

$$\underline{A}\{1,2,4,8\} = \{1,2,4,8\} \quad \underline{A}\{3,5,6,7\} = \{3,7\} \quad \overline{A}\{1,2,4,8\} = \{1,2,4,6,8\} \quad \overline{A}\{3,5,6,7\} = \{3,4,5,6,7,8\}$$

The following are set of *certain rules* in LERS format induced from Table 3 using concept lower approximation

$$2, 2, 2 \quad (\text{Temperature, high}) \ \& \ (\text{Nausea, no}) \ \rightarrow \ (\text{Flu, yes}) \quad (\text{R2.1})$$

$$2, 3, 3 \quad (\text{Headache, yes}) \ \& \ (\text{Nausea, yes}) \ \rightarrow \ (\text{Flu, yes}) \quad (\text{R2.2})$$

$$1, 2, 2 \quad (\text{Headache, no}) \ \rightarrow \ (\text{Flu, no}) \quad (\text{R2.3})$$

The corresponding possible rules set induced from the concept upper approximation is

$$2, 2, 2 \quad (\text{Temperature, high}) \ \& \ (\text{Nausea, no}) \ \rightarrow \ (\text{Flu, yes}) \quad (\text{R2.4})$$

$$1, 3, 4 \quad (\text{Headache, yes}) \ \rightarrow \ (\text{Flu, yes}) \quad (\text{R2.5})$$

$$2, 3, 1 \quad (\text{Temperature, high}) \ \& \ (\text{Nausea, yes}) \ \rightarrow \ (\text{Flu, no}) \quad (\text{R2.6})$$

$$1, 2, 3 \quad (\text{Temperature, normal}) \ \rightarrow \ (\text{Flu, no}) \quad (\text{R2.7})$$

$$1, 2, 2 \quad (\text{Headache, no}) \ \rightarrow \ (\text{Flu, no}) \quad (\text{R2.8})$$

4 A Modified Rough Set Approach to Handle Missing Values

In this approach the definition of indiscernibility relation, used in complete information systems is extended to an equivalent relation [8]. This approach introduces some restrictions and conditions on the similarity relation used to handle incomplete information systems.

Any information table $T = (U, A)$ where U and A are non empty finite sets called the “Universe” and “the set of attributes” respectively. In an incomplete information system some of the attribute values are missing and all these missing attribute values are denoted as “*”. In order to process incomplete information systems the indiscernibility relation has been extended to similarity relation, say $SIM(B)$ which was defined by

$SIM(B) = \{(x, y) \in U \times U, a \in B, f(x, a) = f(y, a) \text{ or } f(x, a) = * \text{ or } f(y, a) = *\}$ and defines a relation $S_B(x)$ as $S_B(x) = \{y \in U : (x, y) \in SIM(B), B \subseteq A\}$, where f is a function that maps the direct product of U and A into the set of all values. $S_B(x)$ denotes the maximal set of objects which are possibly indiscernible by B with x .

Based on the above definitions we can define the reducts of the given decision table and using these reducts the decisions can be taken. But it is evident that when the number of missing values increases in an object it will greatly affect the decision making. But this is not considered in the method discussed in section 3 and hence the generalized decisions are far from the actual decisions. So the number of missing values must be taken into account during the process of analyzing missing values.

To introduce the new definition of similarity relation we use the following definitions of $f(x, a)$, where $f(x, a) = v$ means that the object x has a value v for the attribute a where $x \in U$ and $a \in A$. $f(x, a)$ is called a defined value, if and only if $f(x, a) \neq *$, where $*$ represents a missing value in the given information table. x is called a completely defined object if and only if $f(x, a) \neq * \forall a \in A$. Let $(f(x, a), f(y, a))$ represents a pair of values of the attribute a for the objects x and y respectively. The number EP where $EP = |\{(f(x, a), f(y, a)) \mid \forall a \in B, B \subseteq A\}|$ is the number of equal pairs for the attribute a for all $a \in B$ for all objects x and y respectively. The number, Mx , denotes the number of missing values for the object x . The object x is called well defined object of $B \subseteq A$ iff $Mx \leq N/2$ if N is even or $Mx \leq (N+1)/2$ if N is odd; otherwise x is called poorly defined object. The number NMx denotes the number of not missing values for the object x . Then by considering NMx , an object x is defined as ‘well defined object’ of $B \subseteq A$ iff $NMx \geq N/2$ if N is even or $NMx \geq (N+1)/2$ if N is odd; otherwise it is called poorly defined [8]. Here N represents the number of attributes in B . Now the modified similarity relation (MSIM) can be defined as follows:

- (i) $(x, x) \in MSIM(B)$ where $B \subseteq A, \forall x \in U$;
- (ii) $(x, y) \in MSIM(B)$ where $B \subseteq A, N = |B| \geq 2$ if and only if
 - (a) $f(x, a) = f(y, a) \forall a \in B$ where $f(x, a)$ and $f(y, a)$ are defined values.
 - (b)
$$EP \geq \begin{cases} N/2 & \text{if } N \text{ is even} \\ (N+1)/2 & \text{if } N \text{ is odd} \end{cases} \tag{2}$$

Now, $MS_B(x)$ is defined as $MS_B(x) = \{y \in U : (x, y) \in MSIM(B), B \subseteq A\}$.

Table 4. Incomplete decision table containing missing values only

Case	Temperature	Headache	Nausea	Flu
1	high	*	no	yes
2	very_high	yes	yes	yes
3	*	no	no	no
4	high	yes	yes	yes
5	high	*	yes	no
6	normal	yes	no	no
7	normal	no	yes	no
8	*	yes	*	yes

$MS_B(x)$ is the maximal set of objects which are possibly indiscernible by B with x . If B represents a minimal Reduct of the decision table, then $MSIM(B) = MSIM(A), \forall P \subseteq B, MSIM(P) \neq MSIM(A)$. By constructing the reducts, we can reduce the number of attributes of an information system without affecting the whole information system. By applying the definition of MSIM the vagueness of decisions derived using

SIM approach can be greatly reduced and this leads to a proper decision making even if the decision table contains missing values.

The application of the above definitions on Table 4 with B = A, the complete set of attributes given in Table 4, produces the following results:

Table 5. Complete decision table

Case	Temperature	Headache	Nausea	Flu
1	high	yes	no	yes
2	very_high	yes	yes	yes
3	high	no	no	no
4	high	yes	yes	yes
5	high	yes	yes	no
6	normal	yes	no	no
7	normal	no	yes	no
8	normal	yes	no	yes

$$MSIM(A) = \{(1,1), (2,2), (3,3), (4,4), (4,5), (5,4), (5,5), (6,6), (7,7), (8,8)\};$$

$$MS_A(1) = \{1\} \quad MS_A(2) = \{2\} \quad MS_A(3) = \{3\} \quad MS_A(4) = \{4, 5\}$$

$$MS_A(5) = \{4, 5\} \quad MS_A(6) = \{6\} \quad MS_A(7) = \{7\} \quad MS_A(8) = \{8\}$$

By applying the definition of MSIM (2) the concept B-lower and B-upper approximations are defined as follows:

The concept B-lower approximation $\underline{B} X = \cup\{ MS_B(x) \mid x \in X, MS_B(x) \subseteq X \}$ and

The concept B-upper approximation $\overline{B} X = \cup\{ MS_B(x) \mid x \in X, MS_B(x) \cap X \neq \emptyset \}$.

For the decision table presented in Table 4 and B=A, the available set of attributes, the concept A-lower and A-upper approximations of the two concepts {1, 2, 4, 8} and {3, 5, 6, 7} are:

$$\underline{A} \{1, 2, 4, 8\} = \{1, 2, 8\} \quad \overline{A} \{1, 2, 4, 8\} = \{1, 2, 4, 5, 8\}$$

$$\underline{A} \{3, 5, 6, 7\} = \{3, 6, 7\} \quad \overline{A} \{3, 5, 6, 7\} = \{3, 4, 5, 6, 7\}$$

Rules in LERS format induced from Table 4 using concept approximations are:

The certain rule set:

$$2, 3, 3 \quad (\text{Temperature, high}) \ \& \ (\text{Nausea, no}) \ \rightarrow \ (\text{Flu, yes}) \quad (R3.1)$$

$$2, 2, 2 \quad (\text{Headache, yes}) \ \& \ (\text{Nausea, yes}) \ \rightarrow \ (\text{Flu, yes}) \quad (R3.2)$$

$$1, 2, 2 \quad (\text{Temperature, normal}) \ \rightarrow \ (\text{Flu, no}) \quad (R3.3)$$

The possible rule set:

$$2, 2, 2 \quad (\text{Temperature, high}) \ \& \ (\text{Nausea, no}) \ \rightarrow \ (\text{Flu, yes}) \quad (R3.4)$$

$$1, 3, 4 \quad (\text{Headache, yes}) \ \rightarrow \ (\text{Flu, yes}) \quad (R3.5)$$

$$2, 1, 3 \quad (\text{Temperature, high}) \ \& \ (\text{Nausea, yes}) \ \rightarrow \ (\text{Flu, no}) \quad (R3.6)$$

$$1, 2, 2 \quad (\text{Headache, no}) \ \rightarrow \ (\text{Flu, no}) \quad (R3.7)$$

5 Conclusion

In this paper we discussed three rough set based approaches to deal with missing values. In the first approach, missing values are filled with values found using Core/Reduct concept and a distance measure. The outcome is a complete table. It is observed that the method doesn't give a correct prediction in data sets with multiple match cases. The second approach derives rules based on the available information contained in the incomplete and inconsistent information table based on a Characteristic relation defined from Rough Set theory. In the third approach, the idea used in second approach is modified by considering the number of missing values existing in various cases of the given information table. Rules generated in methods two and three are compared by calculating the confidence of these rules based on the given incomplete decision table and the corresponding complete decision table given in Table 5. It is found that the third approach is slightly better than that of the second approach.

A large number of parameters have been proposed by researchers to ascertain quality of rules generated. The work is being extended by incorporating these parameters as well as large data sets

References

1. Pal, S.K., Skowron, A.(eds.): *Rough Fuzzy Hybridization: A New Trend in Decision Making*. Springer, Singapore (1999)
2. Lin, T.Y., Yao, Y.Y., Zadeh, L.A.(eds.): *Data Mining, Rough Sets and Granular Computing*. Physica-Verlag (2002)
3. Ziarko, W.: *Rough Sets as a Methodology for Data Mining, Rough Sets in Knowledge discovery. Methodology and Applications*, pp. 554–576. Physica-Verlag (1998)
4. Magnani, M.: *Technical Report on Rough Set Theory for Knowledge Discovery in Data Bases* (2003),
<http://magnanim.web.cs.unibo.it/data/pdf/roughkdd.pdf>
5. Duntsch, I., Gediga, G., Nguyen, H.S.: *Rough set data analysis in the KDD process*, <http://citeseer.nj.nec.com/387773.html>
6. Li, J., Cercone, N.: *Assigning Missing Attribute Values Based on Rough Sets Theory*. In: *IEEE Int. Conf. on Granular Computing*, pp. 607–610 (2006)
7. Grzymala-Busse, J.W., Siddhaye, S.: *Rough Set Approaches to Rule Induction from Incomplete Data*. In: *Proc. of IPMU, Italy*, vol. 2, pp. 923–930 (2004)
8. Rady, E.A., Abd El-Monsef, M.M.E., Abd El-Latif, W.A.: *A Modified Rough Set Approach to Incomplete Information Systems*. *J. of Applied Mathematics and Decision Sciences* (2007)
9. *Rough Sets*, http://en.wikipedia.org/wiki/Rough_set
10. Pawlak, Z., Grzymala-Busse, J., Slowinski, R., Ziarko, W.: *Rough Sets*. *Communications of the ACM* 38(11), 89–95 (1995)
11. Li, J., Cercone, N.: *Discovering and Ranking Important Rules*. In: *Proc. of IEEE International Conf. on Granular Computing, Canada*, vol. 2, pp. 506–511 (2005)
12. Grzymala-Busse, J.W.: *Rough Set Theory with Applications to Data Mining. Real World Applications of Computational Intelligence*. Springer, Heidelberg (2004)
13. Little, R., Rubin, D.: *Statistical Analysis with Missing data*, 2nd edn. Wiley, Chichester (2002)
14. Shalabi, L.A., Najjar, M., Kayed, A.A.: *A Frame Work to deal with Missing Data in Data sets*. *Journal of Computer Science* 2(9), 740–745 (2007)

Φ -Rough Sets Theory and Its Usage on Mining Approximate Dependencies

Yiyong Xiao¹, Ikou Kaku², and Wenbing Chang³

¹ Department of System Engineering of Engineering Technology, Beihang University, China
xiaoyiyong@buaa.edu.cn

² Department of Management Science and Engineering, Akita Prefectural University, 84-4
Tsuchiya-Ebinokuti, Honjo, Akita 015-0055, Japan
ikou_kaku@akita-pu.ac.jp

³ Department of System Engineering of Engineering Technology, Beihang University, hina
changwenbing@263.net

Abstract. In this paper, we present an extension of the rough set theory based on similarity relation, called Φ -Rough Set Theory, where the indiscernibility relation of the crisp rough set theory is replaced by the notion of Φ -approximate equivalence relation. Many of the related concepts of crisp rough set theory are redefined, e.g., equivalence class, lower and upper approximations, Φ -reduction, etc. After that, we give an adequate interpretation of approximate dependencies(ADs) and partial approximate dependencies(PADs) from the point of view of Φ -RST, which sheds a new light on the functional dependency(FD) theories. The methodologies of Φ -Rough Set Theory are used on mining the ADs and PADs, on relational or fuzzy relational schema database. Finally, a numeric example is given to illustrate the process of mining the ADs and PADs in decision system database with similarity relation.

Keywords: Rough set theory, Database, Functional dependency, Approximate dependency, Partial approximate dependency.

1 Introduction

In addition to fuzzy sets theory, the rough set theory of Pawlak[1][2] represents a different mathematical approach to vagueness and uncertainty. It has stimulated explorations both on real-world applications and on the theory itself since its emergence in 1982, e.g., in decision analysis, machine learning, knowledge discovery, market research, etc. The classical rough set theory of Pawlak assumes that the information system concerned is a crisp information system based on the crisp relational database, where any feature(attribute) of any object(tuple) has a precise and unique value. Ziarko[3][4] presented the variable precision model of rough set theory(VPRS) to soften or relax the strict constraint in the definition of lower and upper approximations, which leads to this theory to be more competent in dealing with the ill or noisy information. Quafafou[5] transmitted the crisp rough set theory to a fuzzy rough set theory based on fuzzy-set-based fuzzy relational database, where a α -parameterized approximate discernibility relation was employed to replace the crisp discernibility

relation. Chakrabarty et al[6] and Coker[7] incorporate the fuzzy set theory into rough set theory to generate the rough set theory on fuzzy database.

However, in real world, the extents of illness or imprecision of the information on different features of object are always different to somewhat. For some features the values of which can be accurately measured while other features may be far from accuracy. Therefore, besides softening the lower and upper approximations, the discernibility relation, which is the basis of rough set theory, should also be softened to rightly handle fuzziness or imprecision database. The extent of softening the discernibility relations of different fields may be at different levels due to the different extents of imprecision features, which is certain to make the decision system more powerful on handling the imprecision values.

In this paper, we extend the rough set theory by similarity relation on field domain of database, called Φ -Rough Set Theory, in short Φ -RST, where the indiscernibility relation is replaced with the notion of Φ -approximate equivalence class that leads to the redefinitions of all the basic concepts of rough set theory. The Φ -approximate equivalence does not request the tuples in an equivalence class are similar to each other on all attribute at same level, but allows multi-levels of similarity thresholds on different attributes.

After that, we use the methodology of Φ -RST theory on mining approximate dependencies in relational database. The unmatched success achieved by the relational data model[8] in recent decades is essentially due to the concept of relation's simplicity and to its solid theoretical background, i.e., the functional dependencies(FDs). As extensions of FDs, the approximate dependencies(ADs) and partial approximate dependencies(PADs) of relational data model, also attracted extensive attentions[9][10][11]. Mining ADs and PADs on decision database is much attractive both in practice and in academic research, as well as on fuzzy relational database[12][13][14]. In the last section of this paper, we redefined the notions of AD and PAD from the view point of Φ -approximate equivalence to shed a new light on the FDs theories. And then apply the methodologies of Φ -RST to mine the unknown ADs and PADs in a database of decision system. The detailed algorithm is presented, an example of which is given to illustrate the mining process.

This paper is organized as follows: In section 2, we present the Φ -RST based on the Φ -approximate equivalence relation. In Section 3, we first give a redefinition of ADs and PADs based on Φ -approximate equivalence class and interpret the notions of ADs and PADs from the point of view of Φ -RST. Detailed algorithm is given to mine ADs and PADs on decision system database. In Section 4, we give an example of mining ADs and PADs by employing the methodologies of Φ -Rough Set Theory.

2 Φ -Rough Sets Theory

2.1 Φ -Approximate Equivalence Relation

Definition 2.1. α -approximate equivalence relation: For a relational database R, let t_i and t_j be members of R. If t_i and t_j are similar to each other to α level in field c , then t_i is proximate equivalent to t_j with respect to the attribute c with a level value α . Denote it as $t_i \approx_c^\alpha t_j$.

Let $[t_i]_c^\alpha$ represent the equivalence class of tuple t_i with respect to an attribute c with level value α , and $|[t_i]_c^\alpha|$ be the number of elements. Let $\pi_c^\alpha = \{[t_i]_c^\alpha \mid t_i \in R\}$ represent the set of equivalence classes of R under c with level value α , and $|\pi_c^\alpha|$ the number of classes. If not mentioned particularly, the metric $|\cdot|$ always denotes the cardinality of the set in this paper.

Definition 2.2. Φ -approximate equivalence relation: If we have $t_i \approx_{c_k}^{\alpha_k} t_j$ for all $c_k \in C$, $C \subset A$ and $\alpha_k \in \Phi$, then t_i is proximate equivalent to t_j with respect to attributes set C with level values set Φ , where $C = \{c_1, c_2, \dots, c_m\}$, $\Phi = \{\alpha_1, \alpha_2, \dots, \alpha_m\}$, and m is the number of attributes of C . Denote it as $t_i \approx_C^\Phi t_j$.

We can obtain the features of Reflexivity, Symmetry and Transitivity of Φ -approximate equivalence relation.

Lemma 2.1. For $t_1, t_2, t_3 \in R$ such that

$$t_1 \approx_C^\Phi t_1 \quad (\text{Reflexivity})$$

$$t_1 \approx_C^\Phi t_2 \Leftrightarrow t_2 \approx_C^\Phi t_1 \quad (\text{Symmetry})$$

$$t_1 \approx_C^\Phi t_2, t_2 \approx_C^\Phi t_3 \Rightarrow t_1 \approx_C^\Phi t_3 \quad (\text{Transitivity})$$

Let $[t_i]_C^\Phi$ represent the equivalence class of tuple t_i with respect to attributes set C with level values set Φ , and $|[t_i]_C^\Phi|$ be the number of elements. Let $\pi_C^\Phi = \{[t_i]_C^\Phi \mid t_i \in R\}$ represent all the set of equivalence classes of R under attributes set C with level value set Φ , which are called Φ -approximate equivalence classes, and $|\pi_C^\Phi|$ be the number of classes. Some features of the Φ -approximate equivalence class can be induced as the following:

- 1) $[t_i]_C^\Phi \neq \emptyset$
- 2) $R = \cup [t_i]_C^\Phi$,
- 3) If $\alpha \geq \beta$ then $|\pi_C^\alpha| \geq |\pi_C^\beta|$.

Definition 2.3. Φ -Reduction: C is a Φ -Reduction set of attributes if and only if for any $C' \subset C$ and $\Phi' \subset \Phi$ we have $|\pi_{C'}^\Phi| > |\pi_{C'}^{\Phi'}|$, where Φ' and Φ are the similarity thresholds of attributes of C' and C , respectively.

We can get the following Lemmas in accordance to the definition of Φ -Reduction:

Lemma 2.2. For an attribute d and the corresponding similar threshold α such that $|\pi_C^\Phi| \leq |\pi_{C'}^{\Phi'}|$ where $C' = C \cup \{d\}$ and $\Phi' = \Phi \cup \{\alpha\}$.

Lemma 2.3. For an attribute d and the corresponding similar threshold α , d is α redundant if and only if $|\pi_C^\Phi| = |\pi_{C'}^\Phi|$ where $C' = C \cup \{d\}$ and $\Phi' = \Phi \cup \{\alpha\}$.

The redundant attribute is related to its similarity threshold α and the level values set Φ , which may be read as: d is α redundant with respect to attributes set C with level values set Φ .

Lemma 2.4. If an attribute d is α redundant to π_C^Φ , then d is functional dependent to attributes set C with level values set Φ .

2.2 Lower and Upper Approximations

Denote $X \in \pi_D^\Omega$, where π_D^Ω is the set of equivalence classes of R on attributes set D with level values set Ω . D is the decision attributes set and $D \subset A$. $|\pi_D^\Omega|$ is the number of classes. Therefore, the lower and upper approximations of X can be defined consequently:

$$\begin{aligned} \text{Lower approximations:} & \quad \underline{CX} = \{t_i \in R \mid [t_i]_C^\Phi \subset X\} \\ \text{Upper approximations:} & \quad \overline{CX} = \{t_i \in R \mid [t_i]_C^\Phi \cap X \neq \emptyset\} \\ \text{Boundary of X in R:} & \quad \text{BNX} = \overline{CX} - \underline{CX} \end{aligned}$$

With the definitions of lower and upper approximations, X is called a rough set. If an approximate equivalence class defined on C with the level values of Φ is called a granule, i.e., the set: $[t_i]_C^\Phi$ is a granule and π_C^Φ is the set of all granules, then the lower approximation of X is consisted of the tuples of the granules whose elements(tuples) are all contained in X , and the upper approximation of X is consisted of the tuples of the granules that have at least one element (tuple) contained in X . The boundary of X is consisted of the tuples of those granules whose elements(tuples) are partially contained in X .

The notions of \underline{CX} and \overline{CX} above are strictly defined. However, in real world, the data may be contaminated by error or noise. The granule, of which most elements(tuples) are contained in X except only a few ones(which may be ill information or noise), should be assigned to \underline{CX} instead of BNX . On the other hand, the granule, of which most elements(tuples) are not contained in X except only a few ones, should not be assigned to \overline{CX} . Therefore, to identify strong rules, a variable should be added to the rough set theory to calculate \underline{CX} and \overline{CX} , which is called *precision variable* of Φ -rough set theory. Such a notion was firstly presented by Ziarko in his works[3][4]. It is also necessary to add such a variable to Φ -rough set theory to deal with the ill information or noise. Therefore, we get the new definitions of lower and upper approximations of X . For $\beta \in [0.5, 1]$, such that:

$$\text{Lower approximations: } \underline{CX}^\beta = \left\{ t_i \in R \mid \frac{|[t_i]_C^\Phi \cap X|}{|[t_i]_C^\Phi|} \geq \beta \right\},$$

$$\text{Upper approximations: } \overline{CX}^\beta = \left\{ t_i \in R \mid \frac{|[t_i]_C^\Phi \cap X|}{|[t_i]_C^\Phi|} \leq 1 - \beta \right\},$$

$$\text{Boundary of } X \text{ in } R: BNX^\beta = \overline{CX}^\beta - \underline{CX}^\beta,$$

where β is the threshold of the ratio of elements contained in X to those not contained in X , which means that if the ratio of a granule is greater than β then it should be assigned to \underline{CX}^β . Otherwise, it should not. On the other hand, if the ratio is even lower than $1 - \beta$ then the granule should not be assigned to \underline{CX}^β .

Obviously, if $\beta = 1$ then $\underline{CX}^\beta = \underline{CX}$ and $\overline{CX}^\beta = \overline{CX}$, which leads the decision rules found by rough set to be so-called strong rules.

2.3 The Accuracy of Rough Set

The measure of accuracy of a rough set X can be characterized numerically by the following coefficient:

$$\mu(X) = \frac{|\underline{CX}^\beta|}{|\overline{CX}^\beta|},$$

Obviously $0 \leq \mu(X) \leq 1$. If $\mu(X) = 1$, X is crisp or definable; otherwise X is rough with a definability degree of μ . Therefore, the notion of roughness is defined as:

$$\delta(X) = 1 - \mu(X).$$

2.4 Belongingness Matrix

The Φ -approximate equivalence classes are obtained by partitioning the relations R based on the Φ -equivalence relation on condition attributes set C , denoted by π_C^Φ . We define the following coefficient as the belongingness of all these classes with respect to set X :

$$\sigma([t_i]_C^\Phi) = \frac{|[t_i]_C^\Phi \cap X|}{|[t_i]_C^\Phi|},$$

which is the ratio of the number of elements (tuples) that belong to both $[t_i]_C^\Phi$ and X to those of $[t_i]_C^\Phi$. Obviously $0 \leq \sigma \leq 1$, it describes the degree to which a granule ($[t_i]_C^\Phi$) belongs to X .

Besides X , a granule $[t_i]_C^\Phi$ belongs to all the other elements of π_D^Ω to some degrees, the one-to-one belongingness coefficient of the Φ -approximate equivalence classes in π_C^Φ to those in π_D^Ω is a mapping $\sigma_{hr} : \pi_C^\Phi \times \pi_D^\Phi \rightarrow [0, 1]$, which forms a

$|\pi_C^\Phi| \times |\pi_D^\Phi|$ matrix and may be denoted as B , where $\sigma_{hr} \in B, h = 1, 2, \dots, |\pi_C^\Phi|, r = 1, 2, \dots, |\pi_D^\Omega|$. It is easy to get the feature that $\sum_{r=1}^{|\pi_D^\Omega|} \sigma_{hr} = 1$, for all $h = 1, 2, \dots, |\pi_C^\Phi|$.

2.5 Coverage Matrix

In addition to belongingness coefficient, the coverage coefficient is defined correspondently as the following:

$$\rho([t_i]_C^\Phi) = \frac{|[t_i]_C^\Phi \cap X|}{|X|}$$

The coverage coefficient is the ratio of the number of elements(tuples) that belong to both $[t_i]_C^\Phi$ and X to those of X . Obviously $0 \leq \rho \leq 1$, it describes the degree to which a granule($[t_i]_C^\Phi$) covers X .

The one-to-one coverage coefficient of the Φ -approximate equivalence classes in π_C^Φ to the Ω -approximate equivalence classes in π_D^Ω is a mapping: $\rho_{hr} : \pi_C^\Phi \times \pi_D^\Phi \rightarrow [0,1]$, which forms a $|\pi_C^\Phi| \times |\pi_D^\Phi|$ matrix and may be denoted as W , where $\rho_{hr} \in W, h = 1, 2, \dots, |\pi_C^\Phi|, r = 1, 2, \dots, |\pi_D^\Omega|$. It is easy to get the feature that $\sum_{h=1}^{|\pi_C^\Phi|} \rho_{hr} = 1$, for all $r = 1, 2, \dots, |\pi_D^\Omega|$.

2.6 Deterministic Area and Non-deterministic Area

According to the lower and upper approximations defined in **Section 2.2**, the set that unions all the lower approximation is defined as the deterministic area, and the set that unions all boundaries is defined as the non-deterministic, as the following:

$$\Pi = \bigcup_{i=1}^{|\pi_D^\Omega|} \underline{CX}_i^\beta \quad (\text{Deterministic area})$$

$$\bar{\Pi} = \bigcup_{i=1}^{|\pi_D^\Omega|} \overline{BNX}_i^\beta \quad (\text{Non-deterministic area})$$

$$R = \Pi \cup \bar{\Pi}$$

The deterministic area indicates that, for a tuple $t \in \Pi$ the values of the decision attributes are dependent on the values of the condition attributes, which is valuable for decision supporting. The ratio of the deterministic area of the relational schema R can also be characterized by the following coefficient:

$$\lambda = |\Pi|/|R|,$$

which is named as the precision measurement of the decision system $(R, C \cup D)$.

3 On Mining Approximate Dependencies

In this section, we will introduce some basic concepts of approximate dependency firstly, which is an extension of the functional dependencies. And then develop methods of mining approximate dependency or partial approximate dependency by the methodologies of Φ -rough set theory.

3.1 Approximate Dependency

A Function Dependencies(FDs), as defined in[17], is an expression of the type $X \rightarrow Y$ is valid in $R \Leftrightarrow \forall t_1, t_2 \in R$ such that $(t_2.X=t_2.X) \rightarrow (t_1.Y=t_2.Y)$, where $X, Y \subseteq A$. The extensions of FDs, which are called fuzzy functional dependencies(FFDs), have been involved in many proposals. Generally, as defined in[15], a FFD is an expression that $X \rightarrow_f Y$ is valid in $R \Leftrightarrow \forall t_1, t_2 \in R, (\approx(t_1.X, t_2.X) \Rightarrow_f \approx(t_1.Y, t_2.Y))$, where $X, Y \subseteq A, (\approx)$ is a resemblance relation defined on ill-known data aiming to soften or relax the strict equality relation, and (\Rightarrow_f) is a fuzzy implication.

The notion of approximate dependencies(ADs), as mentioned in[16], is an extension of FDs such that equality of tuples is extended and replaced with the notion of approximate equivalence class. If one attribute d is approximately dependent on attributes set C , it means that each approximate equivalence class partitioned with respect to d is functionally dependent on an approximate equivalence class partitioned with respect to C , which can be formulized as the following definition.

Definition 3.1. Approximate dependencies(ADs): An AD over a relational schema A is expressed as $C \rightarrow_a d$, where $C \subset A$, and $d \in A$. Interpreted formally, an AD: $C \rightarrow d$ holds or is valid in a given relation R over A if for any pair of tuples $t_i, t_j \in R$, we have: if $[t_i]_C^\Phi = [t_j]_C^\Phi$ then $[t_i]_d^{\alpha_d} = [t_j]_d^{\alpha_d}$, where Φ is the level values set with respect to C , α_d is the level value of attribute d . Interpreted informally, an AD: $C \rightarrow_a d$ holds if all tuples that agree on C approximately also agree on d approximately.

In additional to FFD, Intan and Mukaidono[11] introduced the concept of partial fuzzy functional dependency(PFFD), which expresses the fact that a given attribute X does not determine Y completely, but in partial area of X it might determine Y . Consequentially, we define the partial approximate dependencies as the following:

Definition 3.2. Partial approximate dependencies(PADs): A PAD over a relational schema A is expressed as $C \rightarrow_{pa} d$, where $C \subset A$, and $d \in A$. Interpreted formally, a PAD: $C \rightarrow_{pa} d$ holds or is valid in a given relation R over A if existing such an approximate equivalence class $[t']_C^\Phi \in \pi_C^\Phi$ that for any pair of tuples $t_i, t_j \in [t']_C^\Phi$ we have: if $[t_i]_C^\Phi = [t_j]_C^\Phi$, then $[t_i]_d^{\alpha_d} = [t_j]_d^{\alpha_d}$, where Φ is the level values set with respect to C , α_d is the level value of attribute d . Interpreted informally, a PAD: $C \rightarrow_{pa} d$ holds if all pairs of tuples of some certain area that agree on C approximately also agree on d approximately.

3.2 Mining Approximate Dependencies by Φ -RST

Rough set describes the relations between two non-overlapped attributes sets while the current functional dependency theory just studies the dependency of one attribute on an attributes set. Therefore, the following theorem and lemma have expressed their relations.

Theorem 3.1. For a decision system $DS=(R,C \cup D)$, an AD $C \rightarrow_a D$ holds or is valid if and only if $C \rightarrow_a d$ holds for all $d \in D$.

Lemma 3.1. For a decision system $DS=(R,C \cup D)$, an AD $C \rightarrow_a D$ holds or is valid if and only if $C \rightarrow_a D'$ holds for all $D' \subset D$.

According to the definition of ADs in Definition 3.1, it is not difficult to prove the theorems and lemma above.

Φ -RST is able to provide a more general formulation to describe the ADs of fuzzy database, as follows:

Definition 3.3. Mining ADs by Φ -RST: An AD over a relational schema A is expressed as $C \xrightarrow{\Phi, \Omega, \beta, \lambda, \Pi} D$, where C is the condition attributes set, D is the decision attributes set, and $A=C \cup D$. Φ and Ω are the level values sets of C and D , respectively. β is the precision variable. λ is the measure of precision of decision system. Π is the deterministic area.

These five parameters of Φ -rough set, namely, $\Phi, \Omega, \beta, \lambda, \Pi$, can systematically describe the approximate dependencies, which are stated in detail as the following:

1. Φ and Ω represent the thresholds of degrees of similarity among the elements(tuples) in a equivalence class. Φ is for the condition attributes and Ω is for the decision attributes, respectively. If all elements of Φ and Ω are equal to 1, then the Φ -approximate equivalence relation degrades to be indiscernibility relation, and the Φ -rough set theory degrades to be a crisp rough set theory.
2. β can be used to soften or relax the strict definition of Lower and upper approximations, which may enhance the ability of resisting ill information or noise in fuzzy database. If $\beta=1$, we call the dependency *strong dependency*; otherwise, *weak dependency*. The degree of weakness can be measured by β .
3. λ can measure the degree of AD: $C \rightarrow_a D$, which raises the notion of partial approximate dependency. We may say the dependency is *approximate dependency* in the case of $\lambda=1$ and *partial approximate dependency* when $0 < \lambda < 1$.
4. Π provides the certain area that the PAD: $C \rightarrow_{pa} D$ holds.

Therefore, the following conclusions can be obtained:

Lemma 3.2. For a decision system $DS=(R,C \cup D)$ on fuzzy database, an AP $C \rightarrow_a D$ holds or is valid if and only if $\lambda = 1$, where λ is the measure of precision of the decision system.

Lemma 3.3. For a decision system $DS=(R,C \cup D)$ based on fuzzy database, a PAD $C \rightarrow_{pa} D$ holds or is valid if and only if $0 < \lambda < 1$, where λ describes the degree to which the PAD holds.

Lemma 3.4. For a decision system $DS = (R, C \cup D)$ based on fuzzy database, a PAD $C \rightarrow_a D$ holds or is valid on the area of Π . Interpreted formally, for any pair of tuples $t_i, t_j \in \Pi$ we have: if $[t_i]_C^\Phi = [t_j]_C^\Phi$, then $[t_i]_D^\Omega = [t_j]_D^\Omega$. Interpreted informally, a PAD: $C \rightarrow_{pa} d$ holds or is valid if all pairs of tuples in Π that agree on C approximately also agree on d approximately.

3.3 Algorithm of Mining Approximate Dependencies

For a relation R over A , where $A=C \cup D$ and $C \cap D=\theta$, the Φ -equivalent classes with respect to C and D are denoted as π_C^Φ and π_D^Ω , respectively. Suppose $\pi_C^\Phi = \{E_1, E_2, \dots, E_l\}$ and $\pi_D^\Omega = \{X_1, X_2, \dots, X_k\}$, where $l = |\pi_C^\Phi|$ and $k = |\pi_D^\Omega|$. We define the measure of belongingness E_i to X_j as:

$$\sigma_{ij} = \frac{|E_i \cap X_j|}{|E_i|},$$

which describes the probability of the tuples contained in E_i being simultaneously contained in X_j . Thereby, the one-to-one belongingness coefficient of the elements in π_C^Φ to those in π_D^Ω is a mapping $\sigma_{ij} : \pi_C^\Phi \times \pi_D^\Omega \rightarrow [0,1]$, which forms an $l \times k$ matrix. Furthermore, we have $\sum_{j=1}^k \sigma_{ij} = 1$, for all $i = 1, 2, \dots, l$.

Proposition 3.1

- 1) If all $0 < \sigma_{ij} < 1$ then attributes set D is independent from C ;
- 2) If all σ_{ij} are either equal to 1 or 0 then $C \xrightarrow{\Phi, \Omega}_a D$ holds;
- 3) If there exist both $\sigma_{ij} = 1$ and $0 < \sigma_{ij} < 1$ then $C \xrightarrow{\Phi, \Omega}_{pa} D$ holds. The areas the PADs hold can be formulated as $\Pi = \left\{ \bigcup E_i \mid \sigma_{ij} = 1 \right\}$, and the degree to which the PADs hold can be formulated as $\lambda = |\Pi|/|R|$.

Below, we introduce the algorithm of mining the Φ -equivalent classes and the ADs/PADs from a relational schema $R = C \cup D$:

Algorithm of finding Φ -equivalent classes and ADs/PADs:

1. **For each** $t_i \in R$
2. $E_i = \phi$ // $E_i \in \pi_C^\Phi$
3. $X_i = \phi$ // $X_j \in \pi_D^\Omega$
4. **For each** $t_j \in R$
5. **If** $t_i \approx_C^\Phi t_j$ **then** $E_i = E_i \cup \{t_j\}$
6. **If** $t_i \approx_D^\Omega t_j$ **then** $X_j = X_j \cup \{t_j\}$
7. **Next For**
8. $\{E_i\} = E_i \cup \{E_i\}$

9. $\{X_j\}=X_j \cup \{X_j\}$
10. **Next For**
11. eliminate the repeated elements in $\{E_i\}$
12. eliminate the repeated elements in $\{X_j\}$
13. **For each** $E_i \in \{E\}$ and $X_j \in \{X\}$
14. $\sigma_{ij} = \text{Card}(E_i \cap X_j) / \text{Card}(E_i)$
15. **Next For**
16. Determining $C \rightarrow_a D$ holds or $C \rightarrow_{ap} D$ holds according to **Proposition 3.1**.
17. If $C \rightarrow_{ap} D$ holds, determine the holding area according to **Proposition 3.1**.

4 Numeric Example

In this section, we give an example to illustrate how to use Φ -rough set theory to mining the approximate dependency in a decision system. A schema of the relational

Table 1. A fuzzy Database Relation(R)

EMP #	JOB	EXP	SALARY
1	Salesman	3	37K
2	Design Engineer	10	40K
3	System Engineer	5	45K
4	Software Engineer	5	45K
5	Accountant	12	47K
6	Accountant	5	50K
7	Secretary	10	53K
8	Secretary	15	55K

Table 2. Similarity relation of *JOB* field

	Sec	SW Eng	Acct	Sys Eng	Sales man	Dn Eng
Sec	1	0.6	0.7	0.6	0.5	0.6
SW Eng	0.6	1	0.6	0.8	0.5	0.8
Acct	0.7	0.6	1	0.6	0.5	0.6
Sys Eng	0.6	0.8	0.6	1	0.5	0.8
Salesman	0.5	0.5	0.5	0.5	1	0.5
Dn Eng	0.6	0.8	0.6	0.8	0.5	1

JOB={Sec,SW Eng,Acct,Sys Eng,Salesman,Dn Eng}

Table 3. Similarity relation of *EXPERIENCE* field

	3	5	10	12	15
3	1	0.9	0.7	0.7	0.5
5	0.9	1	0.7	0.7	0.5
10	0.7	0.7	1	0.9	0.6
12	0.7	0.7	0.9	1	0.6
15	0.5	0.5	0.6	0.6	1

EXPERIENCE={3,5,10,12,15}

Table 4. Similarity relation of SALARY field

	37	40	45	47	50	53	55
37	1	0.9	0.7	0.7	0.5	0.5	0.5
40	0.9	1	0.7	0.7	0.5	0.5	0.5
45	0.7	0.7	1	0.9	0.5	0.5	0.5
47	0.7	0.7	0.9	1	0.5	0.5	0.5
50	0.5	0.5	0.5	0.5	1	0.9	0.9
53	0.5	0.5	0.5	0.5	0.9	1	0.9
55	0.5	0.5	0.5	0.5	0.9	0.9	1
SALARY={37,40,45,47,50,53,55}							

database with similarity relations defined is shown in the Table 1 below representing the JOB, EXPERIENCE and SALARY of eight EMPLOYEES.

The purpose of the decision system is to find the decision rule between JOB, EXPERIENCE and SALARY. So that $R=\{JOB, EXPERIENCE, SALARY\}$, $C=\{JOB, EXPERIENCE\}$, $D=\{SALARY\}$. Suppose $\Phi=\{0.8, 0.7\}$, $\Omega=\{0.9\}$, according to the approximate relation of Definition 2.2, we obtain the following Φ -approximate equivalence classes and Ω -approximate equivalence classes, respectively.

$$\pi_C^\Phi = \pi_{\{0.8,0.7\}}^{\{JOB,EXP\}} = \{\{1\},\{2,3,4\},\{5,6\},\{7\},\{8\}\}, \pi_C^\Phi = \{E_1, E_2, E_3, E_4, E_5\}$$

that $E_1=\{1\}$, $E_2=\{2,3,4\}$, $E_3=\{5,6\}$, $E_4=\{7\}$ and $E_5=\{8\}$.

$$\pi_D^\Omega = \pi_{\{0.9\}}^{\{SAL\}} = \{\{1,2\},\{3,4,5\},\{6,7,8\}\}, \pi_D^\Omega = \{X_1, X_2, X_3\}$$

that $X_1=\{1,2\}$, $X_2=\{3,4,5\}$ and $X_3=\{6,7,8\}$.

Let $\beta=1$, which is the *precision variable* presented by Ziarko[3][4] that will soften the strict constraint definition of lower and upper approximations when β departs from 1 to 0.5, then the lower and upper approximations of $X_i, i=1,2,3$, can be obtained as following:

$$\underline{CX}_1 = \{E_1\} = \{1\}, \overline{CX}_1 = \{E_1, E_2\} = \{1, 2, 3, 4\}, BNX_1 = \{2, 3, 4\}, \mu(X_1) = 0.25$$

$$\underline{CX}_2 = \emptyset, \overline{CX}_2 = \{E_2, E_3\} = \{2, 3, 4, 5, 6\}, BNX_2 = \{2, 3, 4, 5, 6\}, \mu(X_2) = 0,$$

$$\underline{CX}_3 = \{E_4, E_5\} = \{7, 8\}, \overline{CX}_3 = \{E_3, E_4, E_5\} = \{5, 6, 7, 8\}, BNX_3 = \{5, 6\}, \mu(X_3) = 0.5.$$

The belongingness matrix B and the coverage matrix W are:

$$B = \begin{bmatrix} 1 & 1/3 & 0 & 0 & 0 \\ 0 & 2/3 & 0.5 & 0 & 0 \\ 0 & 0 & 0.5 & 1 & 1 \end{bmatrix}, W = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 & 0 \\ 0 & 2/3 & 1/3 & 0 & 0 \\ 0 & 0 & 1/3 & 1/3 & 1/3 \end{bmatrix}.$$

The deterministic area and non-deterministic area are: $\Pi = \{\underline{CX}_1, \underline{CX}_2, \underline{CX}_3\} = \{1, 7, 8\}$, $\overline{\Pi} = \{BNX_1, BNX_2, BNX_3\} = \{2, 3, 4, 5, 6\}$. The measure of precision of the decision system is: $\lambda = |\Pi|/|R| = 3/8$.

According to the Lemma 3.2, we can get the conclusion that the $C \rightarrow_a D$ (or $\{JOB, EXPERIENCE\} \rightarrow_a \{SALARY\}$) does not hold because $\lambda \neq 1$. However, according to

Lemma 3.3 and Lemma 3.4, the $C \rightarrow_{pa} D(\text{or } \{\text{JOB, EXPERIENCE}\} \rightarrow_{pa} \{\text{SALARY}\})$ holds in a degree of $3/8$ and the partial area is just Π , which means that the value of attribute SALARY are partially dependent on the values of JOB and EXPERIENCE in the area of $\{1,7,8\}$.

5 Conclusion

In this paper a new extension of rough set theory based on similarity relation, called Φ -RST, is presented. The approximate equivalence relation: Φ -approximate equivalence relation is defined and replaces the indiscernibility relation of traditional rough set theory, which leads to the redefinition of many notions in Φ -RST. At the same time, the notions of approximate dependencies are also redefined, as well as the partial approximate dependency. Detailed algorithm is given to illustrate the process of using the methodology of Φ -RST to mine approximate dependencies and partial approximate dependencies in a decision system database.

References

1. Pawlak, Z.: Rough Sets. *Int. J. Inf. Comput. Sci.* 11 (1982)
2. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1991)
3. Ziarko, W.: Variable Precision Rough Sets Model. *Journal of Computer and System Sciences* 46(1), 39–59 (1993)
4. Ziarko, W.: Analysis of Uncertain Information in the Framework of Variable Precision Rough Sets. *Foundations of Computing and Decision Science* 18(3/4), 381–396 (1993)
5. Quafafou, M.: α -RST: a Generalization of Rough Set Theory. *Info. Sciences* 124, 301–316 (2000)
6. Chakrabarty, K., Biswas, R.: Fuzziness in Rough Sets. *Fuzzy Sets and Sys.* 110, 247–251 (2000)
7. Coker, D.: Fuzzy Rough Sets are Intuitionistic L-fuzzy sets. *Fuzzy Sets and Sys.* 96, 381–383 (1998)
8. Codd, E.F.: A Relational Model of Data for Large Shared Data Banks. *Communications of the ACM* 13(6), 377–387 (1970)
9. Intan, R., Mukaidono, M.: Fuzzy Functional Dependency and Its Application to Approximate Data Querying. In: *Proceedings of the 2000 International Database Engineering and Applications Symposium* (2000)
10. Tyagia, B.K., Sharfuddin, A., Dutta, R.N., Tayald, D.K.: A Complete Axiomatization of Fuzzy Functional Dependencies using Fuzzy Function. *Fuzzy Sets and System* 151, 363–379 (2005)
11. Hartmann, S., Link, S.: Deciding Implication for Functional Dependencies in Complex-Value Databases. *Theoretical Computer Science* 364, 212–240 (2006)
12. Saxena, P.C., Tyagi, B.K.: Fuzzy Functional Dependencies and Independencies in Extended Fuzzy Relational Database Models. *Fuzzy Sets and Systems* 69, 65–89 (1995)
13. Wang, S.L., Tsai, J.S.: Mining Approximate Dependencies Using Partitions on Similarity-relation-based Fuzzy Databases. In: *Proceedings of the IEEE SMC, Tokyo*, pp. 871–875 (1999)

14. Sozat, M.I., Yazici, A.: A Complete Axiomatization for Fuzzy Functional and Multivalued Dependencies in Fuzzy Database Relations. *Fuzzy Sets and Systems* 117, 161–181 (2001)
15. Yahia, S.B., Ounalli, H., Jaoua, A.: An Extension of Classical Functional Dependency: Dynamic Fuzzy Functional Dependency. *Information Sciences* 119, 219–234 (1999)
16. Wang, S.L., Tsai, J.S.: Discovery of Approximate Dependencies from Proximity-based fuzzy database. In: *Third international conference on Knowledge-based Intelligent Information Engineering System*, Adelaide, Australia (1999)
17. Nakata, M.: Functional Dependencies in Fuzzy database. In: *First International Conference on Knowledge-based Intelligent Electronic System*, Adelaide, Australia, pp. 21–23 (1997)

A Cellular Automatic Method for the Edge Detection of Images

Yu Chen¹ and Zhuangzhi Yan^{1,*}

¹ School of Communication and Information Engineering, Shanghai University
Shanghai, 200072 China
{tzcy, zzyan}@shu.edu.cn

Abstract. This paper present a cellular automaton (CA) based diffusion model and its application in the edge detection of images. The CA-based diffusion model consists of a regular lattice of cells with local state. These cells interact with their neighbors subject to a uniform rule which governs all cells. By setting the initial condition as an image, the diffusion model can be used as an alternative tool for diffusion equation in image processing. Experimental results showed that the CA-based diffusion model has a steady and convergent dynamical behavior and a better performance than the diffusion equation. This model can detects the image edge more accurately and suppress the noise much better than the classical edge detectors, such as LoG, Laplace, Canny and Sobel operators.

1 Introduction

A cellular automaton (CA) is a regular lattice of cells with local state, which interact with their neighbors subject to a uniform rule that governs all cells. The rule is the program that governs the behavior of the systems. It is well known that CA has the potential to perform complex computations with a high degree of efficiency and robustness, as well as to implement massive parallel computing devices. For these reasons CA has been studied and used in image processing. Examples are Wongthanavas [1] and Rosin [2]. However, this approach has significant limitation: The rule is not determined by the global behavior of the CA. Designing CA rule according to the global behavior will lead to a huge rule space, for example in the application of CA to process a binary image, with a Von Neumann neighbor hood, the space of CA rules is as large as 2^{32} . It is quite difficult to find the best rule in such large space.

The challenge posed by the design of CA rule motivates us to explore methods that do not simply rely on the global behavior of the CA. As an alternative, we propose to modify the existent lattice gas CA-based models which are not based on the Boolean operations and have been successfully used for modeling the complex systems, to make use of a predictable dynamics which have the potential for image processing and, therefore, remove the need for seeking the CA rule in a large rule space. To achieve this we examine an approach of the diffusion model, which is adapted to

* Corresponding author.

image processing by setting the initial condition as the image, for example Perona and Malik [3] used the diffusion equation for anisotropic image smoothing while preserving the image edges. We approach this regarding the CA cells as the image pixels and the state of cell as the pixel gray level, deriving a simple expression for the local gradient of image pixels from the lattice gas diffusion model that was proposed by Toffoli and Margolus [4]. Experimental results showed that the modified CA-based model had steady and convergent dynamical behavior while performing better than the diffusion equation. Importantly, compared to the existent CA-based method, this derivation has many applications in image processing because the diffusion equation has been applied to many fields of image processing, for example image enhancement, image segmentation and image recovery etc. In this paper, the modified CA-based model is used for the robust simulation of image edge detection that achieves better performance against the traditional methods such as LoG, Laplace, Canny and Sobel operators. We present quantitative results on synthetic image data and comparative results on MRI image data of human brain.

2 The CA-Based Diffusion Model

We start by considering a lattice gas diffusion model with Von Neumann neighborhood that was proposed by Toffoli and extended by Chopard and Droz [5]. The study of Chopard and Droz showed that in the limit of the infinite lattice, the discrete particle CA model maps directly onto the continuum description, diffusion equation of which the initial condition is equivalent to that of CA model. Koenderink[6] pointed out that when set the initial condition of the diffusion equation as an image, the diffusion equation can be used in image smoothing and the solution of the diffusion equation can be viewed as convolving the original image with a Gaussian kernel. By regarding the cells as the pixels of the image and the particle density as the grey level of the pixel, the CA model can be used as an alternative tool for diffusion equation in image processing.

In our study, we defined the CA rules as following: 1) Each cell of the model connects to its four neighbors by four channels, the particle density equals to the gray level of corresponding image pixel. 2) Each cell can receive no more than 64 particles from each of its neighbors at one moment. 3) At next moment the particles of each cell move out in four directions to the neighbors with probabilities P_0, P_1, P_2 and P_3 , respectively, where $P_0=P_1=P_2=P_3=1/4$, and $P_0+P_1+P_2+P_3=1$.

With these rules, we have developed a modification to the derivation, described in more detail in the Appendix, which provides a formulation for the diffusion given by

$$\frac{\partial \rho(r,t)}{\partial t} = D \cdot \text{div}(\nabla \rho(r,t)) \quad (1)$$

where r is a vector (x, y) denoting the position of the cell. $\rho(r,t)$ is the updated version of the initial image $\rho(r,0)$. Diffusivity D is $D=\lambda^2/(4\tau)$, λ and τ stand respectively for the space and the time intervals, which are described in the Appendix.

From the macroscopic view of point, starting from the initial image $\rho(r,0)$ and by running (1), a family of images $\rho(r,t)$ representing successive versions of $\rho(r,0)$ is constructed. As t increases the image $\rho(r,t)$ changes into more and more simplified

images. The noise is eliminated and the image is smoothed. From the microscopic view of point, starting with the initial image $\rho(r,0)$, the particles randomly move according to the CA rule. During each iteration step, particles move out of the cells and enter the neighbors with probabilities $P_0, P_1, P_2,$ and P_3 in four directions. As iterations increases, the particles are dispersed from the cell of high particle density to the one of low density step by step, and then a smoothed image is obtained.

The choice of D is determining the speed of diffusion. In this research, the space interval λ and the time interval τ are set to be one, and then D is dominated by P_0 . For $P_0=P_1=P_2=P_3$, the largest value of P_0 and D is one fourth. If P_0 is less then one fourth, D will be decreased and the diffusion speed is slow down because some particles are remained where they were. Usually, we can adjust D to be lager than a fourth by adjusting λ and τ . By this means, a D of arbitrary value can be obtained. Condition $P_0+P_1+P_2+P_3=1$, which grantee the conversation of the particles of the CA-based model, is derived from the conversation law.

3 Edge Detection

As an alternative tool to the diffusion equation, the modified CA-based model has more flexible applications than the existent CA method. In this section the modified CA-based edge detection method are described firstly, and then the evaluations are performed. Both qualitative and quantitative evaluations demonstrate the method achieves better performance against the traditional methods such as Sobel, Lapace, LoG, and Canny.

3.1 The CA-Based Edge Detection Method

Consider the equation (1), where $\rho(r,0)$ is the original image. The left side of equation (1) is the second order derivative of the image. For numerical computation, the left side of equation (1) should be discretized, and then the second order derivative of the image $\rho(r, t)$ can be calculated with $[\rho(r,t) - \rho(r,0)]/D$. By finding the zero-crossings, the algorithm can accurately detect the image edges.

To formulate the algorithm, the following initial parameters are defined: 1) The particle number $n_i(r, t)$ moving through the channel i , where $i=1,2,3,4$; 2) The number of iteration K ; 3) A stochastic sequence $u(r,i)$ which denotes the channels selected by particles of each cell such as $\{1,2,4,3,3,4,1,2,3,\dots\}$, the value of which equals 1,2,3,4 with probabilities of 1/4,1/4,1/4,1/4 respectively; 4) Kronecker function $\delta(x)$ which equals 1 if $x=0$ and 0 if $x\neq 0$; 5) The threshold for detecting the edge T ; 6) The total number of the image pixel M .

The algorithm consists of four steps. Fig.1 presents the overview of the algorithm. Step 1: Set $\rho(r,0)$ as the grey level of the image pixel. Step 2: Iterate: a) generate stochastic sequence u on each cell; b) compute $n_i(r,t)=\sum_{j=1}^{\rho(r,t)}\delta(u(r,j)-i)$; c) compute $\rho(r,t+\tau)=n_1(r+\Delta r_1, t) + n_2(r+\Delta r_2, t) + n_3(r+\Delta r_3, t) + n_4(r+\Delta r_4, t)$; d) If $t+\tau \geq \tau K$, then output $\rho(r, t)$ and end the iteration, else $t= t+\tau$; Step 3: Calculate $L(r) = [\rho(r,t) - \rho(r,0)] / (D \cdot t)$. Step 4: Search the zero-crossings with the thresholding value $T=\sum L(r)/(2M)$.

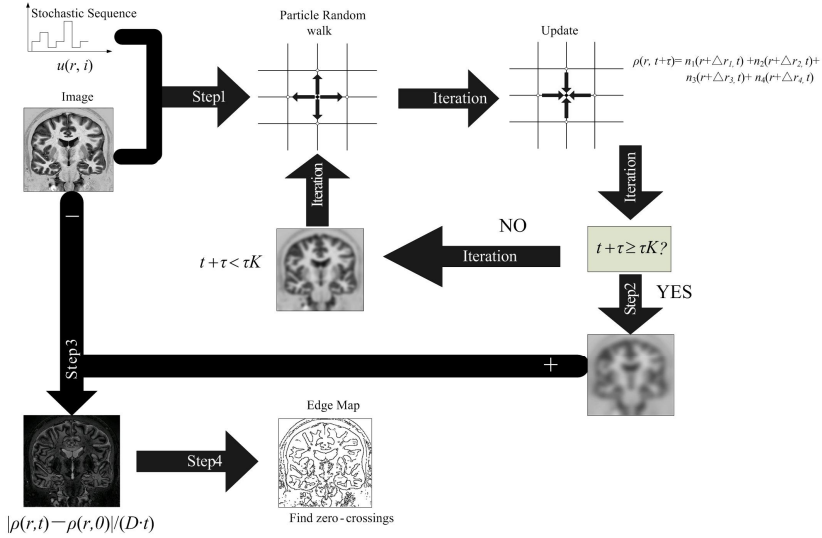


Fig. 1. A schematic overview of the algorithm

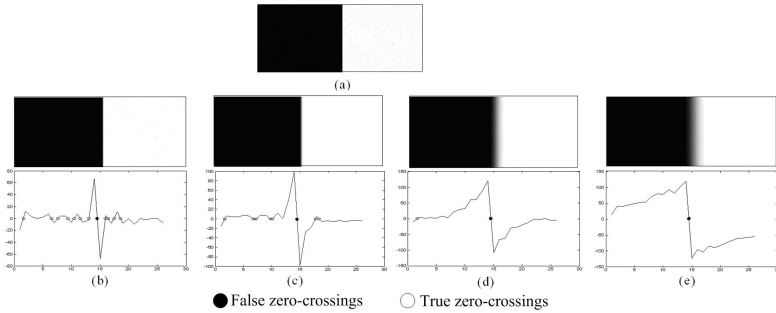


Fig. 2. Illustration of the behavior of the CA-based edge detection method on a synthetic noisy image

In this study, $[\rho(r, t) - \rho(r, 0)]/D$ is used to calculate the second order derivative of the image $\rho(r, 0)$. But the false zero-crossings would degrade the edge map. We eliminate the false zero-crossings by adjusting the parameter t . Fig. 2(a) is the synthesized noisy image with a Gaussian noise of variance $\sigma = 2$. Fig. 2(b) to Fig. 2(e) illustrate the $\rho(r, 1)$, $\rho(r, 5)$, $\rho(r, 100)$, $\rho(r, 200)$ of Fig. 2(a), and the central part of the cross-section of line number 150 of $[\rho(r, t) - \rho(r, 0)]/(D \cdot t)$, where $t = 1, 5, 100$ and 200 . With the increasing of t , the number of the false zero-crossings is dramatically reduced and the dislocation is small.

3.2 Qualitative Evaluation

To qualitatively demonstrate the effect of the CA-based edge detection method on real image, we compared the result processed by the modified CA-based model with

those by some classical edge detectors. Fig. 3 shows the edge detection using the modified CA model, LoG, Laplace, Canny and Sobel operators. Fig. 3(a) is the original MR image of human brain’s coronal plane; Fig. 3 (b) to Fig. 3(f) are the edge detected images resulting from the CA-based model, LoG, Laplace, Canny, and Sobel operators, respectively. The modified CA-based model clearly detects almost the entire image features, such as the cerebral longitudinal fissure, the third ventricle, the pons, the white matter, and the medulla oblongata. The edge is continuous and the false zero-crossings are suppressed. Compared with the CA method, Fig. 3(c) has too much breaks and the edges are not clear, Fig. 3(d) which is processed by Laplace operator contains too much noise, the edges of Fig. 3(f) are too thick, Fig. 3(e) is much better but it still has some noise and breaks.

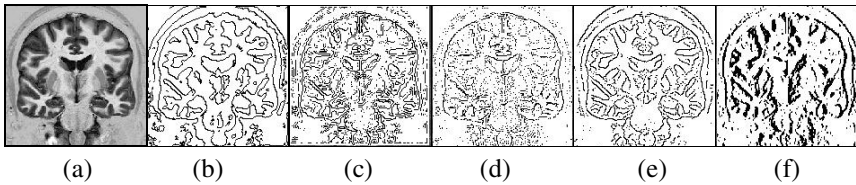


Fig. 3. Edges resulting from different operators (a) Original image; (b) CA-based model; (c) LoG operator; (d) Laplace operator; (e) Canny operator; (f) Sobel operator.

3.3 Quantitative Evaluation

In order to evaluate the performance of the modified CA method quantitatively in edge detection, we calculate an index E for evaluating the edge detection results detected on an array of 259-by-259 images which are generated artificially. The index E , proposed by Pratt [7], was computed as:

$$E = \sum_{i=1}^{I_A} \left(\frac{1}{1 + \alpha \cdot d(i)^2} \right) / \max(I_A, I_I),$$

where I_A is

the number of edge pixels detected by the edge detector; I_I is the actual number of edge pixel in the test image; $d(i)$ is the distance between the actual pixel and the one found by the edge detector. α is a constant used for scaling, and as it was in Pratt’s work, a value of 1/9 is used. This index is a function of distance between the actual and detected edge pixels. Better the edge detection is, closer to one the E is, and E will be one if the detected edges are exactly the actual edges. In this study, a group of test images are generated by adding white noise with different variance to Fig. 4. Fig. 8 is the graph of E resulting from compared edge detectors calculated from the test

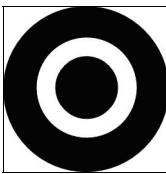


Fig. 4. Original test image

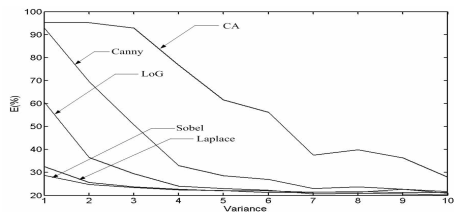


Fig. 5. The graph of E resulting from compared edge detectors

images. As shown in Fig. 5, the CA-based method provides the highest E , while the other detectors give lower E values while the noise variance increases. This means the edge found by the CA-based model is more close to the actual edge, and can suppress the noise much better than the other operators.

4 Discussion

4.1 Uniform and Non-uniform Transition Rules

The development of CA has left a wrong impression that CA should be restricted to a regular lattice and should use uniform transition rules during its evolution. The capabilities of computers were reduced by these apparent restrictions. But, this does not mean people should use the non-uniform transition rules in all of their studies. What kind of rules should be used is determined by what kind of model is to be described. In this research, we used a uniform transition rule to model the diffusion model. The main reason is that, from a physical point of view, the particles of the diffusion phenomenon in real world all move randomly according to a same law. Thus, the uniform transition rules can accurately describe the diffusion model. Moreover, this research is the base-ment of the further study for designing non-uniform CA rules in advanced image processing. A non-uniform CA rule can be designed easily by modifying the parameters of our model. For example, considering the diffusivity D is determined by the parameters P_i , people can modify P_i to be variables of the image gradient, and therefore a non-uniform transition rule is designed. Such a non-uniform CA rule has the potential to be used in anisotropic image diffusion. This is what should be studied in our future work.

4.2 The Dynamical Behavior of the CA-Based Model

In image smoothing, starting with the initial image $\rho(r,0)$, the particles randomly move according to the CA rule. During each iteration step, particles move out of the cells and enter the neighbors with probabilities P_0, P_1, P_2 and P_3 in four directions. As iterations increases, the particles are dispersed from the cell of high particle density to the one of low density step by step and a smoothed image is obtained. When parameter t is large enough, the modified CA-based model is reaching a steady state. Particle densities of all the cells are almost the same and the image is totally blurred. This means the modified CA-based model is steady and convergent in image smoothing.

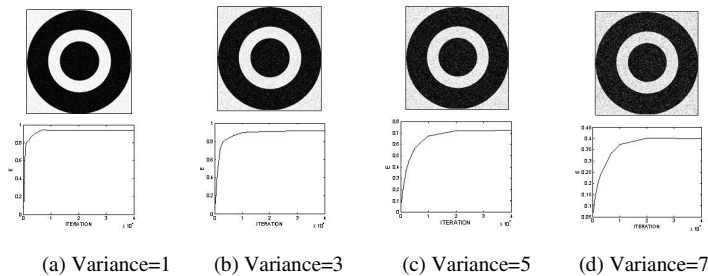


Fig. 6. The dynamical behavior of the CA-based edge detection method

For edge detection, we used the parameter E to investigate the dynamical behavior of the modified CA-based model. Fig.6 shows the dynamical behavior of the CA-based method, the variance of noise in Fig.6 (a) to Fig.6 (e) equates to 1, 3, 5, and 7, respectively. With the increasing of the iterations, the E values tend to approach to a constant which do not change much when the variance of noise has a minor variety. Fig.6 illustrates that the CA-based method is steady and convergent in edge detection.

5 Conclusions

In this paper we propose to modify the existent lattice gas CA-based diffusion model which has been successfully used for modeling the complex systems, to make use of a predictable dynamics which have the potential for image processing and, therefore, remove the need for seeking the CA rule in a large rule space.

This modified CA-based model has many applications in image processing, for example edge detection. By computing the second order derivative and finding the zero-crossings, the edge map of the image is obtained. In the qualitative evaluation, we qualitatively evaluated the performance of the modified CA-based model by comparing it with the traditional edge detectors, such as LoG, Laplace, Canny and Sobel operator, in detecting the edge of a MR image of human brain's coronal plane. In the quantitative evaluation of the modified CA-based model, the method proposed by Pratt was used to quantitatively evaluate the results of the detection. The evaluation shows that the modified CA-based model is steady and convergent and also provides the highest E , while the other detectors give lower E s when the variance of noise increases. This means the edge found by the CA-based model is more close to the actual edge and can suppress the noise much better than the other operators.

There are a few interesting problems worth for further investigation. Firstly, in edge detection, the number of iteration K is chosen manually. With a larger K , the model will probably lose some important edges; with a smaller K , the model will detect more false edges which are caused by the false zero-crossings. Therefore, an automatically chosen iteration number will improve the performance of the model in edge detection. Secondly, considering the diffusivity D is determined by the parameters P_i , P_i can be modified to be variables of the image gradient, and thus a non-uniform transition rule is designed. It is interesting and useful to design such a non-uniform CA rule which has the potential to be used in anisotropic image diffusion.

References

1. Wongthanavas, S., Sadananda, R.: A CA-based Edge Operator and Its Performance Evaluation. *J. Vis. Comm. and Image Representation* 14(2), 83–96 (2003)
2. Rosin, P.L.: Training Cellular Automata for Image Processing. *IEEE T. Image Process* 15(7), 2076–2087 (2006)
3. Perona, P., Malik, J.: Scale-Space and Edge Detection Using Anisotropic Diffusion. *IEEE T. Pattern Anal.* 12(7), 629–639 (1990)
4. Toffoli, T., Margolus, N.H.: *Cellular Automata Machines: A New Environment for Modeling*. MIT Press, Cambridge (1987)

5. Chopard, B., Droz, M.: Cellular Automata Model for the Diffusion Equation. J. Statistical Physics 64(3-4), 859–892 (1991)
6. Koenderink, J.: The Structure of Images. Biol. Cybern. 50(5), 363–370 (1984)
7. Abdou, I.E., Pratt, W.K.: Quantitative Design and Evaluation of Enhancement/Thresholding Edge Detectors. Proc. IEEE 67(5), 753–763 (1979)

Appendix: Derivation of the Diffusion Equation (1)

The particle number entered into the cell in position $(r+\lambda c_i)$ at next time step $t+\tau$ from direction i ($i=1,2,3,4$) can be written as:

$$n_i(r+\lambda c_i, t+\tau) = \sum_{m=1}^{n_{i+0}(r,t)} \mu_{m,i+0}(r,t) + \sum_{m=1}^{n_{i+1}(r,t)} \mu_{m,i+1}(r,t) + \sum_{m=1}^{n_{i+2}(r,t)} \mu_{m,i+2}(r,t) + \sum_{m=1}^{n_{i+3}(r,t)} \mu_{m,i+3}(r,t) \tag{2}$$

where $\mu_{m,i+j}(r,t)$ ($j=0,1,2,3$) is a Boolean variable that denotes the m^{th} particle that entered into the cell in position r from direction $j+i$ at time t and move out in direction $i+2$ at time $t+\tau$. It equals 1 with probability P_j and 0 with probability $(1-P_j)$. Take the expectations of the both sides of equation (2), we have

$$N_i(r+\lambda c_i, t+\tau) = P_0 N_i(r,t) + P_1 N_{i+1}(r,t) + P_2 N_{i+2}(r,t) + P_3 N_{i+3}(r,t) \tag{3}$$

where $N_i(r,t)$ is expectation of $n_i(r,t)$. Expand $N_i(r,t)$ as $N_i(r,t) = N_i^{(0)} + \varepsilon N_i^{(1)} + \varepsilon^2 N_i^{(2)} + \dots$, where ε is an infinitely small term. Substitute $N_i(r,t)$ for the right side of equation (3) and expand the left side of equation (3) in an infinite Taylor series, we obtain

$$\left[\tau \frac{\partial}{\partial t} + \frac{\tau^2}{2} \left(\frac{\partial}{\partial t} \right)^2 + \lambda \left(c_i \cdot \frac{\partial}{\partial r} \right) + \dots \right] N_i(r,t) = \sum_{k=0}^{+\infty} \left[\left((P_0 - 1) N_i^{(k)} + P_1 N_{i+1}^{(k)} + P_2 N_{i+2}^{(k)} + P_3 N_{i+3}^{(k)} \right) \right] \varepsilon^k \tag{4}$$

Use the Chapman-Enskog condition: $\lambda \sim \varepsilon \lambda$, $\tau \sim \varepsilon^2 \tau$ and $\rho = N_1^{(0)} + N_2^{(0)} + N_3^{(0)} + N_4^{(0)}$, and consider about the $O(\varepsilon^0)$, $O(\varepsilon^1)$ and $O(\varepsilon^2)$ of both sides of (4), thus

$$N_i^{(0)} = \rho / 4 \tag{5a}$$

$$N_i^{(1)} = -\lambda \left(c_i \cdot \frac{\partial}{\partial r} \right) N_i^{(0)} \tag{5b}$$

$$\sum_{i=1}^4 \left[\tau \frac{\partial}{\partial t} N_i^{(0)} + \lambda \left(c_i \cdot \frac{\partial}{\partial r} \right) N_i^{(1)} + \frac{\lambda^2}{2} \left(c_i \cdot \frac{\partial}{\partial r} \right)^2 N_i^{(0)} \right] = 0 \tag{5c}$$

respectively. The diffusion equation $\frac{\partial \rho(r,t)}{\partial t} = D \cdot \text{div}(\nabla \rho(r,t))$, where $D = \lambda^2 / (4\tau)$, can be derived by substituting both (5a) and (5b) in (5c).

Computer Virus Evolution Model Inspired by Biological DNA

Yu Zhang, Tao Li, and Renchao Qin

College of Computer Science, Sichuan University, Chengdu 610065, China
bullzhangyu@yahoo.com.cn

Abstract. There has been considerable interest in computer viruses since they appeared. Firstly, computer viruses make software developers pay more attention to security and develop anti-virus technology more carefully. Secondly, computer viruses are vital simulation platforms to study biological viruses. Moreover, computer viruses (a form of artificial life) are helpful to enhance the understanding of the philosophy of life. Therefore, computer viruses and their evolutions are worth thoroughly studying in scientific research sense. To further investigate the evolution of computer viruses, a DNA-based model for computer viruses evolution, which draws inspiration from biological DNA and artificial life, is proposed. The formal definition of computer virus is introduced, and the evolution operators which include selection, mutation, and recombination operator are presented. The simulation experiments were conducted and the results indicate that computer viruses have enormous capabilities of self-propagation and self-evolution.

Keywords: computer virus evolution, biological DNA, artificial life.

1 Introduction

The ease of Internet information transmission and the vulnerabilities of computer systems provide good opportunities for computer viruses propagation [1, 2, 3]. So far, computer viruses have caused enormous economic losses. In the endless arms race between viruses and anti-viruses, anti-virus technologies are generally in disadvantageous position. At present, most mature anti-virus technologies are based on viruses' signatures and subsequently they have advantages in detecting known viruses, not in detecting unknown viruses or mutations of known viruses. In order to enhance the detection rate of unknown viruses or mutations of known viruses, a further study on how computer viruses evolve must be carried on, which makes anti-viruses effectively detect unknown viruses so as to protect information systems against damage [4]. This is called "Know yourself and know your enemy, victory is assured."

Computer viruses are generally considered by people as malicious codes that only consume system resources, block network, and damage data. As a possible form of artificial life, however, computer viruses may do good to the study of biological viruses because of some similarities between them such as self-replication, self-construction, and self-evolution. Therefore, how computer viruses evolve may have enlightenment in

the deeper understanding of the mechanistic properties of biological viruses, which could contribute to improvements in public health, and play a unique and positive role in exploring the natural origin of life and the essence of life on earth [5, 6].

So far, there is little literature in the computer virus evolution domain. Ludwig [7] discussed the relationship between computer virus and artificial life and their evolution. Agapow [8] examined the allegations that computer viruses are improbable to evolve due to programs computational brittleness, and concluded that programs are far less brittle than expected. Hao [5] discussed the evolution of computer viruses from the philosophical angle. SPTH [9] introduced the possibility of computer viruses evolution based on the similarity between program code and biological DNA. They studied computer viruses and their possible evolution, but none of them investigated how computer viruses evolved. To further investigate computer viruses evolution and how they evolve, a DNA-based model which draws inspiration from artificial life and biological DNA is proposed. The structure of computer viruses, the life cycle of computer viruses, and the relationship between computer viruses and biological ones are introduced. This model comprehensively simulates the propagation of biological viruses, and profoundly reveals the evolution of computer viruses. Experimental results show that computer viruses, which have both algorithm characteristics and life characteristics, have extremely enormous evolution capabilities.

In the remaining sections of the paper, we first presented the proposed model in Section 2. Experiments and results are discussed in Section 3. Conclusions are stated in Section 4.

2 The Proposed Model

2.1 The Life Cycle of Computer Viruses

Biological viruses hijack a biological cell and make it produce many new viruses from a single original. They have evolved elaborate mechanisms of cell infection, proliferation and departure from the host when it bursts from virus overcrowding. Similarly, a computer virus is a program that is able to infect other programs by modifying them to include a possibly evolved copy of it. The likeness between computer virus and biological virus makes computer virus a possible form of artificial life, which can self-replicate and self-evolve during the proliferation to infected machines. The life cycle of computer viruses can be divided into four stages, namely, the generation of new computer viruses, the propagation of computer viruses, the triggering of computer viruses and the eradication of computer viruses, as shown in Figure 1. And Figure 1 implies that computer virus evolution which caused a variety of computer virus variants generally occurs in the stage of computer viruses' propagation.

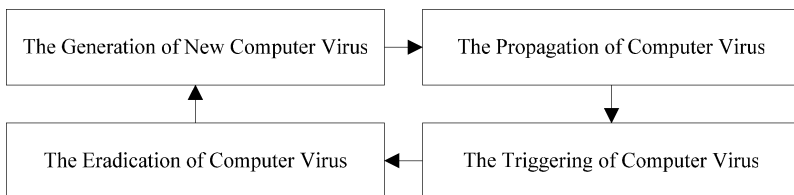


Fig. 1. The life cycle of computer virus

2.2 The Logical Structure of Computer Viruses

The self-replication of computer programs, the vulnerabilities of computer systems, and the sharing of computer networks are the primary foundation of computer viruses evolution, which consequently determines the logical structure of computer viruses. Generally, a computer virus functionally consists of infection mark, initialization module, propagation module, and performance module. The infection mark indicates whether or not the host program is infected; the initialization module makes preparation for viruses execution; the propagation module makes its copy infect the other programs as more as possible; the performance module does annoying things such as causing random damage to data files, occupying disk space and main memory, and using up CPU processing time.

Based on the computer virus structure discussed above, we can formally define it as a quaternion $V = (V_1, V_2, V_3, V_4)$, in which V_1 labeled the set of infection marks, V_2 the set of initialization modules, V_3 the set of propagation modules, and V_4 the set of performance modules. Each set also consists of n -vector, namely, $V_i = (v_{1i}, v_{2i}, \dots, v_{ni})^T$. Thus, the computer virus can be represented as the following equation,

$$V = \begin{pmatrix} v_{11} & v_{12} & v_{13} & v_{14} \\ v_{21} & v_{22} & v_{23} & v_{24} \\ \vdots & \vdots & \vdots & \vdots \\ v_{n1} & v_{n2} & v_{n3} & v_{n4} \end{pmatrix} \tag{1}$$

The computer virus structure is so reasonable as to make full use of system resources to carry out sabotage activities [10, 11].

2.3 The Relationship between Biological DNA and Computer Virus Code

DNA[12] is the chromosomal component composed of *Adenine* (A), *Thymine* (T), *Cytosine* (C) and *Guanine* (G) that carries hereditary information. Similarly, the computer virus program is composed of functions, and functions consist of codes which are composed of commands built of the two bits (0, 1). Consequently, we can obtain the following equations which represent the relationship between a biological cell and a computer virus program.

$$\text{Chromosome} = \text{Program Function} \tag{2}$$

$$\text{DNA} = \text{Code} \tag{3}$$

$$\text{Genes} = \text{Commands} \tag{4}$$

$$\text{Bases} = \text{Bits} \tag{5}$$

Firstly, a chromosome is built of many DNAs. It has its own functions and is responsible for specifically jobs. Similarly, programs consist of functions. These functions are responsible for their own jobs. Secondly, the biological DNA is responsible for the development and the behavior of a cellular life form. Equally, the code of a program is responsible for the development and the behavior of the program. Thirdly, a gene has one small specifical function, and it is build of bases. It is nearly the same as our command which has a small function, and consists of bits (0, 1). The slightly

difference is that the roots of the chromosome are the four bases, but the roots of programs are two bits. Based on the relationship between a biological cell and a computer virus program mentioned above, we naturally have an idea that computer viruses may be evolve in the cyberspace just as the lives evolve in the nature.

Inspired by the similarity between a biological cell and a computer virus program mentioned above, we can represent a computer virus as a DNA-based chromosome, which was proposed by Equation (1). As a result, the variation of its evolution could be a Descartes set $V_{\text{evolution}}$, namely,

$$V_{\text{evolution}} = \prod_{i=1}^4 V_i = V_1 \times V_2 \times V_3 \times V_4 = \{(v_1, v_2, v_3, v_4) \mid \forall i \leq 4, v_i \in V_i\} \quad (6)$$

2.4 The Computer Virus Evolution Model

Due to the algorithm characteristics and life characteristics, the viruses' mutations or unknown viruses are generally created in the evolution of known computer viruses. Virus writers have created new viruses as the following steps. Firstly, the known viruses are code-analyzed. Secondly, various modules are extracted within the viruses. Finally, a new virus is generated by using a variety of algorithms to combine known virus modules. The main steps are shown in Figure 2.

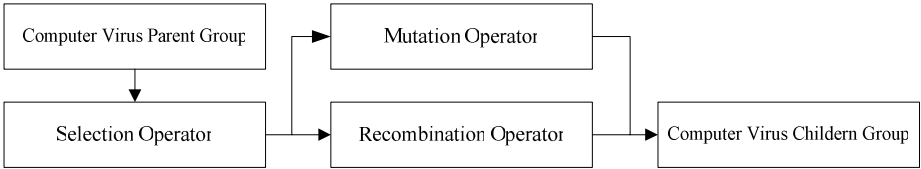


Fig. 2. The evolution of computer viruses

We will mathematically model the three operators including selection operator, mutation operator and recombination operator, which are inspired by the biological viruses' evolution on earth.

2.4.1 The Selection Operator

$$V_{\text{select}}(t) = \begin{cases} V_{\text{initial}}, & t = 0 \\ V_{\text{select}}(t-1) - V_{\text{del}}(t), & t \geq 1 \end{cases} \quad (7)$$

$$V_{\text{del}}(t) = \{v \mid v \in V_{\text{select}}(t-1) \wedge f_{\text{adaption}}(v) = 0\} \quad (8)$$

$$f_{\text{adaption}}(v) = \begin{cases} 0, & \text{detected} \\ 1, & \text{otherwise} \end{cases} \quad (9)$$

Equation (7) depicts the computer virus evolution process using the selection operator, in which V_{initial} is a virus initialization set, and V_{del} is the virus set detected and

removed by anti-virus. Virus fitness $f_{adaption}$ is defined as a piecewise function, whose value is 0 if detected by anti-virus, otherwise 1. This model simulates the selection process of genetic algorithm, that is, the virus with highly fitness is selected from the current virus groups to constitute the mating pool of the next generation viruses.

2.4.2 The Mutation Operator

$$V_{mutation}(t) = \begin{cases} V_{select}, t = 0 \\ V_{mutation}(t-1) - V_{del}(t) + V_{new}(t), t \geq 1 \end{cases} \quad (10)$$

$$V_{del}(t) = \{v \mid \exists v \in V_{mutation}(t-1) \wedge (\exists x \in V_{sig}, Match(x, v) = 1)\} \quad (11)$$

$$V_{new}(t) = \{v \mid \exists x \in V_{mutation}(t-1) \wedge (v \in Mutation(x))\} \quad (12)$$

$$Mutation(x) = \{x \mid x.v_i \xrightarrow{p} x.v_j, 1 \leq i, j \leq 4\} \quad (13)$$

$$Match(x, y) = \begin{cases} 1, \exists i, 1 \leq i \leq 4, x.v_i = y.v_i \\ 0, otherwise \end{cases} \quad (14)$$

Equation (10) depicts the computer virus evolution process using the mutation operator, in which V_{select} is the mating pool from the selection process, V_{del} is the deleted virus set for being matched by any of signatures in antivirus signatures set V_{sig} during mutation, and V_{new} is the new generation individuals through mutation operation. Equation (13) simulates mutation operation process to generate the new virus, in which a randomly selected virus exchanges its genes with a certain probability p to generate the new viruses. Equation (14) is the matching function between computer virus and antivirus signatures, whose value is 1 if matched by antivirus signatures, otherwise 0. Figure 3 shows the mutation operator for example. One command can change to another one with mutation of only one bit of it.



Fig. 3. The illustration of mutation operator

2.4.3 The Recombination Operator

$$V_{recombination}(t) = \begin{cases} V_{select}, t = 0 \\ V_{recombination}(t-1) - V_{del}(t) + V_{new}(t), t \geq 1 \end{cases} \quad (15)$$

$$V_{del}(t) = \{v \mid \exists v \in V_{recombination}(t-1) \wedge (\exists x \in V_{sig}, Match(x, v) = 1)\} \quad (16)$$

$$V_{new}(t) = \{v \mid \exists x, y \in V_{recombination}(t-1) \wedge (v \in Recombination(x, y))\} \quad (17)$$

$$Recombination(x, y) = \{(x, y) \mid x.v_i \leftrightarrow y.v_j, 1 \leq i, j \leq 4\} \quad (18)$$

$$Match(x, y) = \begin{cases} 1, \exists i, 1 \leq i \leq 4, x.v_i = y.v_i \\ 0, otherwise \end{cases} \quad (19)$$

Equation (15) depicts the virus evolution process using the crossover operator, in which V_{select} is the mating pool from the selection process, V_{del} is the deleted virus set because of being matched by any of signatures in antivirus signatures set V_{sig} during recombination, and V_{new} is the new generation individuals through recombination operation. Equation (18) simulates recombination operation process to generate the new virus, in which two randomly selected viruses mutually exchange genes in the seat of the corresponding alleles to generate the new viruses containing more complex genes. Figure 4 shows the mutation operator for example. The new commands are generated by recombination of three different commands.

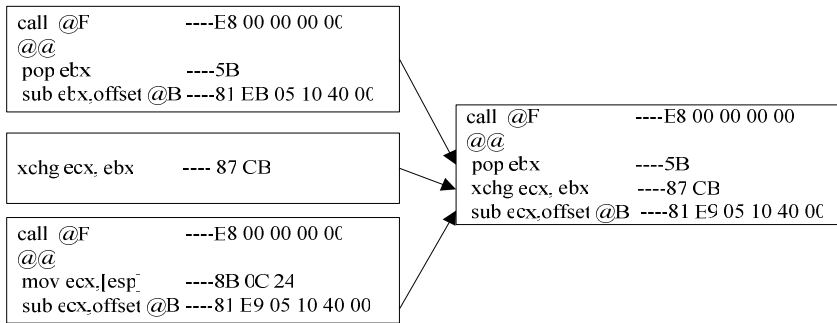


Fig. 4. The illustration of recombination operator

3 Experiment and Results

The experiments were carried out in the computer virus and anti-virus laboratory at computer network and information security Institute of Sichuan University. The goal of the experiment is to evaluate the evolution of computer viruses and how computer viruses evolve with the proposed model. Ten Windows script viruses' samples were chosen from the WildList [13]. Experimental environment is shown in table 1.

We did experiments to test our model by watching how many viruses can survive during the evolution. To simplify the experiment, each virus was only extracted its performance module to form the final virus gene pool which is totally composed of 10

Table 1. Experiment platform of computer virus evolution

– Experiment platform	– VMWare5.0, Windows XP,2.4GHZ /512M	
– Anti-virus tool	– Kaspersky7.0	
– The number of virus genes	– 10	
– Windows script virus samples	– VBS/Soraci, – VBS/Redlof.C, – VBS/Solow!ITW#10, – VBS/Yosenio!ITW#1, – VBS/Gedza, –	– VBS/Areses!ITW#48, – VBS/Solow!ITW#1, – VBS/Solow!ITW#20, – VBS/Z0X!ITW#1, – VBS/Redlof.A-m

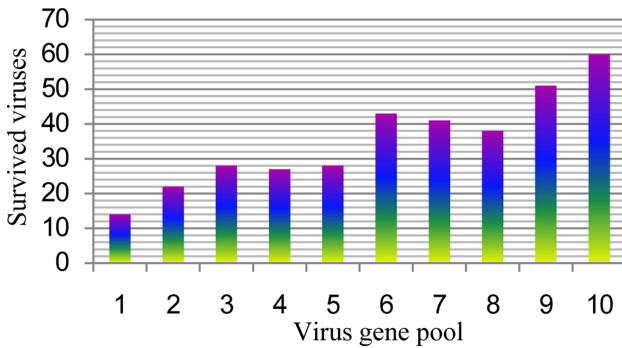


Fig. 5. The relationship between virus genes and survival viruses

virus genes. This experiment was divided into 10 groups. A randomly selected virus gene was in the first group, and two randomly selected viruses genes were in the second group, and so on. All of ten virus genes were in the 10th group. Each experiment has generated 100 virus samples.

The experimental results are shown in Figure 5. There is an obvious tendency that the survived new virus generations are proportional to the virus genes during the evolution. Theoretically, with the increasement of virus samples, the virus gene pool also will be larger, which results in avoiding premature convergence and loss of diversity within the virus population. Thus, in the process of the virus evolution, the new viruses’ chromosomes have a diversity of genes, which expanded their signature space, and thus increasing the probability of their survival. Suppose that m randomly selected genes out of the virus gene pool consisting of n genes are reorganized, then there are P_n^m different types of new viruses produced in the evolution. Since there are tens of thousands of viruses at present, the computer viruses have enormously evolutionary capabilities with the proposed evolution operators.

4 Conclusions

Computer viruses, on one hand, have always been an important research topic in the information security community since they are helpful to enhance anti-virus technology; on the other hand, they are very important research objects in the study of artificial life. Based on the study of computer viruses and their logic structure and life characteristics, a DNA-based model for computer virus evolution is proposed. This model comprehensively simulates the computer virus propagation process, and profoundly reveals how computer viruses evolve. The experimental results show that computer viruses have enormously evolutionary capabilities. Meanwhile, this model provides research idea for anti-virus to take precautions to combat with virus. Anti-virus that applies gene analysis to detect virus gene could achieve the purpose of intelligently eradicating unknown viruses. In addition, this model could cast a new and intriguing light on how artificial life propagate and evolve, and provide new ideas and inspirations for the study of the biological virus evolution.

Acknowledgements. This work was sponsored by the National Natural Science Foundation of China (60573130), National 863 project of China (2006AA01Z435), and the New Century Excellent Expert Program of Ministry of Education of China (NCET-04-0870). The authors acknowledge the support of computer virus and antivirus Lab of Sichuan University.

References

1. Richard, F., Spafford, E.H.: Happy Birthday, Dear Viruses. *Science* 317(5835), 210–211 (2007)
2. Balthrop, J., Forrest, J., Newman, M.E.J., Williamson, M.M.: Technological Networks and the Spread of Computer Viruses. *Science* 304, 527–529 (2004)
3. Lloyd, A.L., May, R.M.: How Viruses Spread among Computers and People. *Science* 292, 1316–1317 (2001)
4. Nachenberg, C.: Computer Virus-antivirus Coevolution. *Communications of the ACM* 40(1), 46–51 (1997)
5. Hao, N.X.: Computer Virus: a Possible Form of Life (2005), <http://cyborg.bokee.com/2586799.html>
6. Spafford, E.H.: Computer Viruses—a Form of Artificial Life? Technical Report, Purdue University (1994)
7. Ludwig, M.A.: Computer Viruses, Artificial Life and Evolution. American Eagle Publications, Tucson (1993)
8. Agapow, P.M.: Computational Brittleness and the Evolution of Computer Viruses. *Parallel Problem Solving from Nature — PPSN IV* (1996) 1141, 1–11 (1996)
9. SPTH.: Code Evolution: Follow nature's example (2005), <http://vx.netlux.org/lib/vsp10.html>
10. Cohen, F.: Computer Viruses, PhD thesis, University of Southern California (1985)
11. Szor, P.: The Art of Computer Virus Research and Defense. Symantec Press (2005)
12. Watson, J., Crick, F.: A Structure for Deoxyribose Nucleic Acid. *Nature* 171, 737–738 (1953)
13. The WildList Organization International (2008), <http://www.wildlist.org/>

Study of Immune Control Computing in Immune Detection Algorithm for Information Security

Ya Jing Zhang¹ and Yang Xue²

¹ Institute of Computer Science & Technology, Yantai University, Yantai, Shan dong, 264005, China

² Faculty of Electric and Automatic Engineering, Shanghai University of Electric Power, Shanghai, China

yajing_zh@163.com, xueyang@shiep.edu.cn

Abstract. Based on research for anomaly detection and immune control mechanism in the biological immune principles, a method of immune control computing is introduced in immune detection. Such problems as immune detection model and immune memory are discussed. The experiments that are in the area of intrusion detection are made. Shown by experiments, the algorithm joined immune control computing provides a dynamic detecting way and keeps stability of detection. Its detecting efficiency is increased obviously.

Keywords: immune control; immune detection; information security; vaccines.

1 Introduction

In essence, information security is the valid control of information. The mechanism of active defense is a difficult problem in the area of information security. The biological immune system is successful at protecting the human body against a vast variety of foreign pathogens [1]. The artificial immune system inspiring by biological immune principles is a good solution hitherto. Immune detection algorithms in artificial immune system have gained prominence because of their promise to provide for feasible and efficient detection mechanisms.

Negative selection algorithm proposed by Forrest and Hofmeyr is based on negative selection method in biological immune mechanisms [2]. However, it shows several drawbacks. The major limitation appears to be the excessive computational time caused by the random generating the initial repertoire. Moreover, it is very difficult to know whether the number of generated detectors is large enough that can satisfy the accepted detection failure probability. P. D'haeseleer provides the two algorithms: The linear time algorithm and the greedy algorithm [3]. The former has made it practical to construct efficient detector sets for large data sets. The space requirements for this construction algorithm are substantial. However, this space is only needed once, to calculate the detector set, and can be discarded afterwards. The later sacrifices some of the speed of detector generation in exchange for a more compact detector set and a failure probability guaranteed to be below the acceptable Pf. Kim and Bentley adopt such a strategy as a clone selection operator with negative selection operator for

network intrusion detection [4]. Immune control computing begins with 1990's. Bersini, Krishnakuma and Takahash are the main researchers. But study of immune control is usually discussed in the control area.

This paper based on real date sets to study the feasibility of immune selection investigates the use of immune control and vaccines operator provided by an immune selection algorithm within an artificial immune system. In order to solve the scaling problem of an independent negative selection algorithm, a niching strategy and vaccines operator are adopted in this paper.

2 Biological Immune Control Method

Immune system controls the individual responses by B-cells and T-cells. The process of immune feedback control is shown in Fig.1 [4]. By interplay between main feedback and inhibition feedback, immune system can rapidly respond to foreign material and stabilize itself simultaneously.

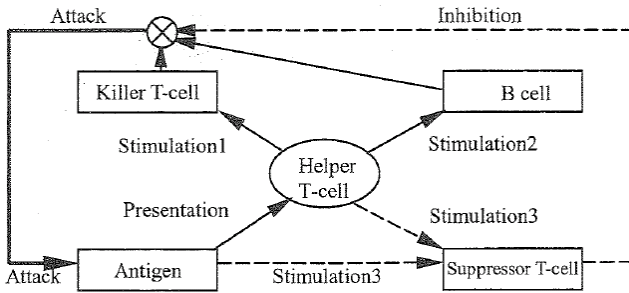


Fig. 1. Scheme of the whole immunity Circle

3 Immune Detection Algorithm

3.1 Negative Selection Algorithms

Forrest et al [1] proposed and used a negative selection algorithm for various anomaly detection problems. This algorithm defines 'self' by building normal behavior patterns of a monitored system. It generates a number of random patterns that are compared to each self pattern defined. If any randomly generated pattern matches a self pattern, this pattern fails to become a detector and thus it is removed. Otherwise, it becomes a 'detector' pattern and monitors subsequent profiled patterns of the monitored system. During the monitoring stage, if a 'detector' pattern matches any newly profiled pattern, it is then considered that new anomaly must have occurred in the monitored system.

3.2 Clone Selection Algorithm with Negative Selection Operator

In order to solve the scaling problem of an independent negative selection algorithm, Kim and Bentley [4] argued the artificial immune system that adopts a clonal selection

algorithm which embeds a negative selection operator. A conclusion is made that the embedded negative selection operator plays an important role in the artificial immune system by helping it to maintain a low false positive detection rate.

3.3 Gene Immune Detection Algorithm

Gene immune detection algorithm (GIDA) [5] is mainly based on such characteristics as high detecting efficiency and low false alert efficiency in biological immune system and the most important function of immune system, that is, which is able to alert and opposes to novel attacks.

The main idea is that vaccine mechanism and positive selection is added when the clone selection method is combined with negative selection to obtain a comprehensive definition of the nonself space.

4 Immune Control Model and Immune Control Computing

4.1 Mathematical Model of Immune Control

The Th-cells concentration relate to the inbreaking antigens concentration. If we define $Th(k)$ as the Th-cells concentration and $Ag(k)$ as the inbreaking antigens concentration, this relation can be presented by Hill's function:

$$Th(k) = f_1(Ag(k)) \tag{1}$$

$$f_1(Ag(k)) = \frac{Ag^2(k)}{(\theta^2 + Ag^2(k))} \tag{2}$$

Where $Ag(k)$ stands the concentration of the k " generation antigen, θ^2 is a range constant and concerned with the kinds of antigen and Th-cell.

The k th generation Ts-cells concentration is concerned with the difference of the d th forth generation antibody, then the k th generation Ts-cells concentration can be presented by:

$$Ts(k) = k_1 f_2(\Delta Ab(k - d)) \cdot Ag(k) \tag{3}$$

$$\Delta Ab(k - d) = Ab(k - d) - Ab(k - d - 1) \tag{4}$$

Where k_1 is rate constant, d denotes the life cycle of antibody, $Ts(k)$ stands for Ts-cells concentration.

We adopt the nonlinear function as follows:

$$f_2(\Delta Ab(k - d)) = 1 - \exp\left\{-\left[\Delta Ab(k - d)\right]^2 / a\right\} \tag{5}$$

Where $a \in (0,1)$.

Th-cells activate B-cells product antibodies, but Ts-cells suppress B-cells product antibodies. So the antibodies concentration can be presented by the difference of Th and Ts concentration:

$$Ab(k) = Th(k) - Ts(k) \tag{6}$$

Substituting (I), (2) and (3) into (6):

$$\begin{aligned} Ab(k) &= f_1(Ag(k)) - k_1 \cdot f_2(\Delta Ab(k - d)) \\ &= \frac{Ag^2(k)}{\theta + Ag^2(k)} k_1 \cdot \left\{ 1 - \exp\left[-\frac{[\Delta Ab(k - d)]^2}{a}\right] \right\} \cdot Ag(k) \end{aligned} \tag{7}$$

If we define the governing error $e(t)$ in control system as an antigen $Ag(k)$, the output $u(t)$ as antibody $Ab(k)$, we can transform (7) to (8):

$$\begin{aligned} u(t) &= f_1(e(t)) - k_1 \cdot f_2(\Delta u(t - d)) \\ &= \frac{e^2(t)}{\theta + e^2(t)} k_1 \cdot \left\{ 1 - \exp\left[-\frac{[\Delta u(t - d)]^2}{a}\right] \right\} \cdot e(t) \end{aligned} \tag{8}$$

The configuration of immune system is shown by Fig.2.

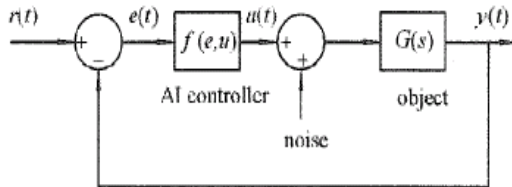


Fig. 2. Configuration of immune control system

4.2 Immune Control Computing

In this paper, on the basis of the previous research, an immune detection algorithm with immune control computing (ICDA) is discussed. In this algorithm, according to the biological immune method, after negative selecting, a B-cells set generates. In order to obtain detection sets (antibodies), the system generates an antigens set in advance. When it encounters antigens in the first time, the B-cell with high fitness value will be stimulated to be an antibody and cloned to be new more antibodies. This is help T-cells mechanism. At the same time, the suppressor T-cells mechanism starts up. Through the niching strategy, all the antibodies but one with high similarity will be killed, the antibodies library generates. As the antigens intrude, the antibodies can recognize antigens relying on muzzy match. The antigens with high fitness will be feedback to the antibodies library and to be vaccines. Those antibodies which have not been used will be killed. Thus, the new generation of antibodies occurs.

The flow diagram of ICDA is shown in Figure 3

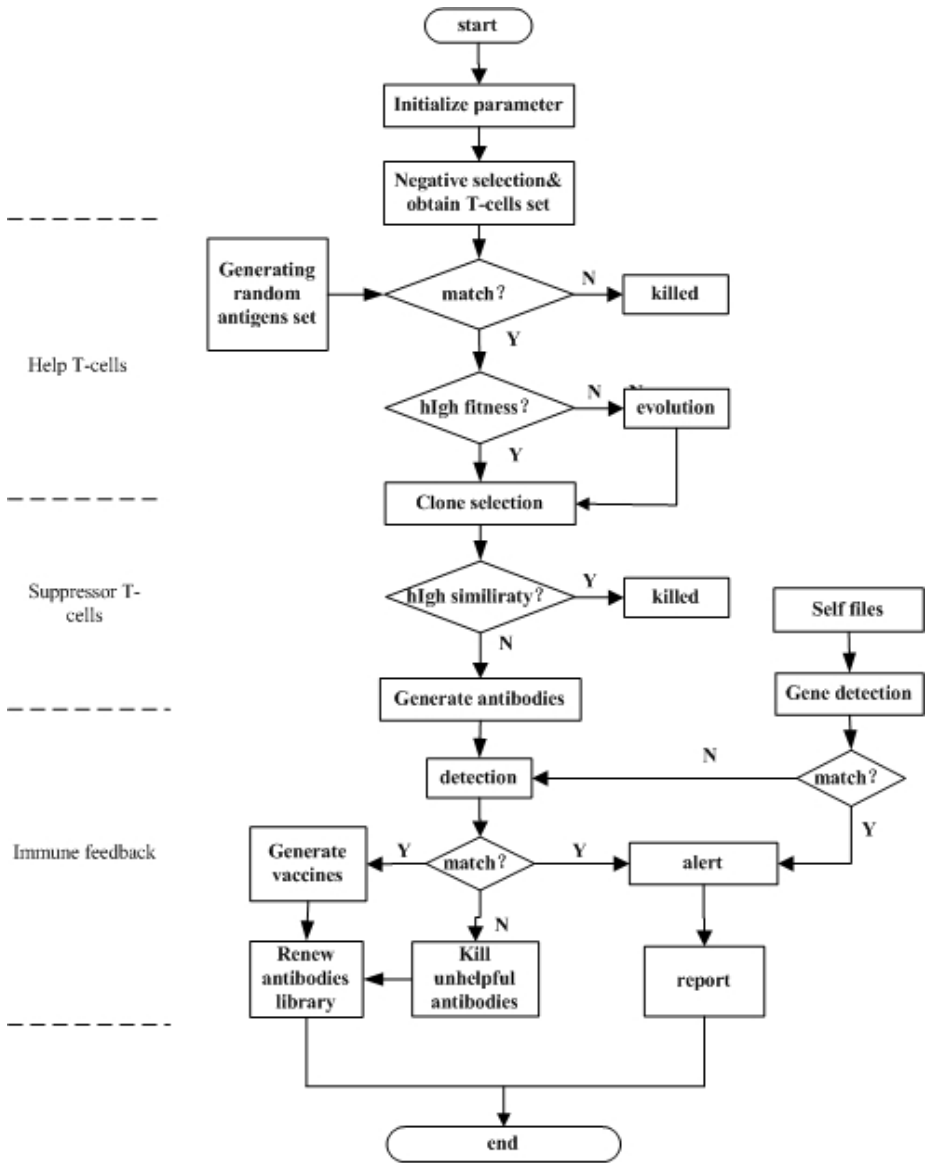


Fig. 3. The flow diagram of Immune Control Detection Algorithm

4.3 Immune Memory

In our algorithm, the process of extracting vaccines is a very important. There are two methods of vaccination: using expert’s previous knowledge rule-matches data files; during the censoring process, vaccines are generated by algorithm itself and the

antibodies responding at the first time join antibodies library as vaccines. Then detecting data files protected using positive selection operator. Once matched, it will be considered as illegal.

Here, the vaccines can also be taken as memory cells that immune system generates after it responds to antigens encountered at the first time in the process of immunity. The function of memory cells is that the immune system is able to respond rapidly through activating memory mechanism when the antigen is encountered again. Under the r-contiguous match rule, the accurate definition of vaccine is a scheme which achieves r bit gene contiguous match. Thereby, a vaccine isn't a complete individual, but some characteristics of the individual depend on the situation of the genes. But because of continuous bit match, a vaccine is asked for a fixed place. Thus, the antibody including the vaccine joins vaccine-antibodies library as the carrier of the vaccine. Based on the character of negative selection, the generating process of vaccines is as followed: 1) Generate nonself set using negative selection algorithm. 2) Detect nonself. 3) The antigens detected join vaccine-antibodies library as vaccines. The principle of extracting vaccines lies in affirming the detector that may be repeated as a vaccine-antibody. Thus, it either guarantees vaccines' quality or increases antibodies' diversity.

In the following study, the immune control method will be improved according to the biological mechanisms and control methods. The immune controller could be used so as to improve the algorithm's shortcomings.

Table 1. Detection and alarm for the intrusion behaviours

Testing item	Operation step	Anticipation result
1. modifying Password	a. acquire authorization of root; b. modifying password file: vi/etc/password; c. add a back door user with the following command lines: newuser: X: 00::/home/newuser: /bin/bash; d. save and exit.	The number of antigens detected is shown, alarm.
2. setting script SUID bit	a. acquire authorization of root; b. setting SUID bit: chmod -perm-4000/bin/tcsh; c. owner of file is instead of root: chown root tcsh; d. save and exit.	The number of antigens detected is shown, alarm.
3. modifying the important file self-defined by user	a. acquire authorization of root; b. modifying the important file self-defined by user; c. save and exit.	The number of antigens detected is shown, alarm.
4 modifying host computer's log file	a. acquire authorization of root; b. landing with a new user name, modifying host computer's log file c. save and exit.	The number of antigens detected is shown, alarm.

Table 2. Contrast of GIDA and ICDA (intrusion detection experimental Result)

Attacking mode	GIDA			ICDA			Ls	Lsa
	NR	Time	Pf	NR	Time	Pf		
Modifying Password	67	0.241	0.130	60	0.408	0.034	2006	1975
	90	0.301	0.091	88	0.525	0.050		
	103	0.409	0.041	69	0.291	0.031		
Setting SUID bit of script	110	0.204	0.350	78	0.507	0.060	288604	288604
	178	0.569	0.203	131	0.799	0.122		
Modifying important files self-defined by user	89	0.017	0.255	56	0.00308	0.090	14	11
	125	0.046	0.148	70	0.00234	0.115		
	179	0.087	0.097	104	0.00536	0.026		
Modifying master computer's log files	58	0.104	0.062	28	0.0409	0.110	2351	1657
	86	0.312	0.010	72	0.0760	0.037		

The data in Table 2 shows the feasibility of GIDA and ICDA in the area of intrusion detection;

Ls (bytes)= the Length of source file, Lsa (bytes) =the Length of the file after attacked; NR = the number of detectors, Time (s)= detecting time , Pf =the failure probability of detection.

When the file protected is long enough, i.e. in the item of Setting SUID bit of script, it is unnecessary that protecting the whole file. Here, we adopt the method of sampling the file. The head, tail and midst of the file protected will be sampled to be a selfish file. Based on such characteristics as distributed and specific, at the time of sampling the middle information of the source file, a dynamic method that sampling in random every time is used. We can verify that if the sampling situation has appropriate distributing probability and sampling times is enough, the result is creditable.

5 Experiments

Based on the above analyses, a kind of experiment is conducted in order to check the effectiveness and feasibility of the proposed algorithm: intrusion detection. The parameters are set as followed: m—the number of alphabet symbols, $m=2$, l—the number of symbols in a string, $l=32$, r—the number of contiguous matches required for a match, $r=8$.

The environment of hardware: two test computers. One is used to imitate attack actions. Another is used to detecting intrusion or computer virus. The environment of software: Windows XP, Linux, IE explorer, VC++ 6.0.

The process of the experiment is shown as Table 1.

The result of the experiment is shown as Table 2.

In order to show the effect of ICDA, GIDA is made to be compared with it.

6 Conclusion

In this paper, we have described a novel change detection algorithm---immune control detection algorithm (ICDA). The immune control method is introduced in the first. The process of detection is regard as a system. The stimulation and compression of T-cells embody the extent of immune control. Immune feedback control makes the system obtain stability. System information can be feedback and detected in time. Thus, the detection method can be considered as a dynamic detecting way. Shown by experiments, it is very effective for the change detection of the compiled C object file. The efficiency of detection is increased and the number of valid detectors is largely decreased.

The key to the approach lies in the realization of immune control method. We imitate the function of help T-cells and suppress T-cells in the immune control. Feedback procedure is designed as a dynamic state. A variety of detection operators is joined in the process.

References

1. Marrack, P., Kappler, J.W.: How the immune system recognizes the body. *Scientific American* 269(3), 81–89 (1993)
2. Forrest, S., Perelson, A.S., Allen, L.R., et al.: Self-nonsel Self Discrimination in a Computer. In: *Proceedings of the 1994 IEEE Symposium on Research in Security and Privacy*, pp. 271–281. IEEE Computer Society Press, Los Alamitos (1994)
3. D'haeseleer, P., Forrest, S., Helman, P.: An Immunological Approach to Change Detection: Algorithms, Analysis, and Implications. In: *Proceedings of the 1996 IEEE Symposium on Computer Security and Privacy*, pp. 98–109 (1996)
4. Chen, W.Q., Zhou, J.Z., et al.: Compensatory Controller Based on Artificial Immune System. In: *International Conference on Mechatronics and Automation*, pp. 1608–1613 (2006)
5. Zhang, Y.J., Yang, X., Wu, S.H.: A Gene Immune Detection Algorithm with a Strategy of DNA PRI. In: *2004 International Conference on Machine Learning and Cybernetics*, pp. 476–481 (2004)

A Comparative Study of Machine Learning Methods for Detecting Promoters in Bacterial DNA Sequences

Leonardo G. Tavares, Heitor S. Lopes*, and Carlos R. Erig Lima

Bioinformatics Laboratory
Federal University of Technology Paraná (UTFPR),
Av. 7 de setembro, 3165 80230-901, Curitiba (PR), Brazil
leonardo.tavares@up.edu.br, hslopes@pesquisador.cnpq.br,
erig@utfpr.edu.br

Abstract. Machine Learning methods have been widely used in bioinformatics, mainly for data classification and pattern recognition. The detection of genes in DNA sequences is still an open problem. Identifying the promoter region laying prior the gene itself is an important aid to detect a gene. This paper aims at applying several Machine Learning methods to the construction of classifiers for detection of promoters in the DNA of *Escherichia coli*. A thorough comparison of methods was done. In general, probabilistic and neural network-based methods were those that performed better regarding accuracy rate.

Keywords: Bioinformatics; Data classification; Machine Learning; DNA.

1 Introduction

The DNA (Deoxyribonucleic acid) is a organic molecule responsible for the coordination and functioning of all living beings. It not only carries the genetic information of the organism, but also stores all the necessary information for synthesizing proteins.

The DNA is composed by simple elements (monomers) forming a long polymer. Monomers, in turn, are called nucleotides and are formed by chemical compounds and one of four types of molecules (bases): adenine (A), cytosine (C), guanine (G) and thymine (T).

Within the cell, the DNA is organized into structures called chromosomes, and the set of chromosomes is the genome of the organism. Most part of the DNA of eukaryotes (organisms having a membrane surrounding the nucleus of their cells) is not expressed in an amino acid chain of a protein is known as "junk DNA". For instance, only 1.5% of human genome is known to be protein-coding [16]. Although it is speculated that junk DNA may be involved in regulatory or catalytic activities, their function is still unknown. The remaining segments

* This work was partially supported by the Brazilian National Research Council – CNPq, under research grant no. 309262/2007-0 to H.S. Lopes.

of DNA that are related to the genetic information of the organism are called genes. Many effort has been spent in developing methods for identifying genes in DNA, from both Biochemistry/Biology and Computer Science communities.

Proteins, the product of genes, are synthesized after transcription and traduction of stretches of DNA. The beginning of the transcription of the gene takes place when the enzyme RNA-polymerase binds to given regions of the DNA known as promoters. These regions indicate to the enzyme that the genetic information to be transcribed is about to come in the linear sequence of nucleotides. Therefore, the following stretch of DNA is transcribed into mRNA (messenger Ribonucleic Acid), from which introns will be later extracted, and then will map a protein [7]. The rules that dictate the behavior of RNA-polymerase are not precisely known. Therefore, many studies have been carried out on this subject, basing on known examples. Using such approach, supported by Computational Intelligence methods, researchers aim at improving the current knowledge of this important biological process.

After the genome of an organism be sequenced, the following task is the identification of genes present in its DNA. Actually, gene detection is still an open problem and many methodologies have been proposed. The most usual approach for finding a gene is detecting signals, that is, particular sequences with biological meaning in the DNA. Among them, searching for promoter regions is one of the most important tasks [9], since a promoter indicates precisely where a gene will start in the DNA sequence.

Currently, the techniques most frequently used for recognizing promoter regions in DNA are based on machine learning methods, such as neural networks, decision trees and others.

Machine learning is a branch of Computer Science related to the development of algorithms and techniques that allow computers to learn from data. Basically, there are two types of computational learning: inductive and deductive. The later type is focused on logic. On the other hand, the methods of the former type are the most popular and they extract rules or formatted knowledge from data. The objective of this work is to evaluate and compare several machine learning methods applied to the detection of promoter regions in the DNA of a common bacteria, *Escherichia coli*.

2 Methodology

2.1 The Data Set

In this work we use a data set compiled by Towell et al. [13] with 106 instances, half of which having promoter sequences (positive cases) and the remaining being intragenic sequences (negative cases). The positive cases were withdrawn from another data base created by Harley and Reynolds [5], whereas the negative cases were obtained by selecting contiguous substrings from a 1500-base sequence (for more details, see [13]).

Each instance is 57 nucleotide long, and promoter regions correspond to positions -50 to +7 from the beginning of a gene. That is, the 50 nucleotides that

precede the starting point of a gene, together with the 7 first nucleotides of the gene. For simplicity purposes, in this work we did not consider positions from +1 to +7, considering only the nucleotides that precede the gene.

2.2 Computational Methods

Two computational tools for machine learning were used in this work. The first software is known as Weka [15] and is frequently used by the data mining community. It has a large number of conventional methods for data analysis in a single tool. HMMER version 2.3.2 [2] is the other software, and is a tool used by the bioinformatics community for the analysis of sequences.

The input file for Weka was formatted to ARFF model. This is a text file in which attributes are described and corresponding instances are listed. The convention used for data was: $m_{50}, m_{49}, \dots, m_1$, corresponding to positions from -50 to -1 in the nucleotide sequence, followed by the target attribute, that identifies the class of each instance.

HMMER was used for evaluating the Hidden Markov Model (HMM) method. HMMs have been used for pattern recognition in many domains and, in special, in bioinformatics for the detection of patterns in protein and DNA sequences [2, 14]. HMMs are considered in this work due to its efficiency in dealing with the probabilistic nature of biological sequences. A HMM is an statistical method considered a simple Bayesian network. A system modeled by a HMM is considered as a Markovian process which parameters are unknown. Consequently, the objective is to estimate the value of the parameters from known instances, and then, use the model for analyzing unknown instances and detecting patterns [3].

The input file for HMMER must be a pre-aligned set of sequences. The 53 positive sequences (that is, corresponding to a promoter sequence) were already aligned in the original database.

3 Computational Experiments and Results

A total of 30 different methods for supervised learning in Weka were used. These methods were grouped as Bayesian, neural networks, meta-learners, trees, lazy-learners and rules.

The default training/testing methodology included a 10-fold cross-validation [6]. That is, the data set is divided into 10 parts. In the first round, nine parts are used for training and the remaining for testing. Next, another other nine partitions are chosen for training and one is set apart for testing. This procedure is repeated until all 10 possible combinations have been tested. The reported result is the average of the 10 runs. The purpose of cross-validation is to avoid biased results when using a small sample of data.

Each HMM was constructed using the tool *hmmbuild* and, later calibrated using *hmmcalibrate* [2]. In both cases the running parameters were configured to the standard values. The detection threshold was empirically set to -8.4, in such a way to minimize the number of errors.

The main objective of this work is to compare the performance of classifiers in detecting promoters. Therefore, performance is measured according to the predictive accuracy of the classifiers and other parameters, such as processing time or memory requirements were not taken into account.

Table 1 shows the results for the 31 supervised classification methods. Methods are divided by category and, within categories they are ordered by descending order of predictive accuracy. In this table it is also shown parameters used in the evaluation of supervised classifiers and, from which, some metrics can be derived. These parameters are drawn from a confusion matrix (or contingency table). For a two-class prediction problem, outcomes can be binary labelled as positive (class "promoter") or negative (class "non-promoter"). When applying a given classifier method to a set of instances, depending on the outcomes, four different parameters can be computed:

- *tp*: true positive - number of positive instances that were correctly classified as positive;
- *fn*: false negative - number of positive instances that were wrongly classified as negative;
- *fp*: false positive - number of negative instances that were wrongly classified as positive;
- *tn*: true negative - number of negative instances that were correctly classified as negative.

By combining these parameters, it is possible to compute several metrics commonly used in machine learning, such as sensitivity (*Se*), and specificity (*Sp*), defined in Eq. 1. Another measure of quality for two-class problems is the Matthews Correlation Coefficient (MCC) (Eq. 2), regarded as a balanced measure and frequently used in bioinformatics [8, 11]. These measures are also shown in Table 1 for all classifiers.

$$Se = \frac{tp}{(tp + fn)} \quad Sp = \frac{tn}{(tn + fp)} \quad (1)$$

$$MCC = \frac{tp \cdot tn - fp \cdot fn}{\sqrt{(tp + fp) \cdot (tp + fn) \cdot (tn + fp) \cdot (tn + fn)}} \quad (2)$$

A ROC (Receiver Operating Characteristics) graph is a useful technique for comparing classifiers and observing visually their performance. This kind of graph is commonly used not only in decision making, but also in machine learning, data mining and bioinformatics [12]. In a ROC graph axes *x* and *y* are defined, respectively, as $1 - specificity$ and *sensitivity*. These axes can be interpreted as the relative trade-offs between the benefits and costs of a classifier. In this work, all classifiers were run once since they used the standard parameters. Therefore, the ROC graph can be represented by a single ROC point for each non-parametric classifier, corresponding to their $(1 - Sp, Se)$ pairs [4].

When comparing classifiers using a ROC graph, the best possible prediction method would be that lying as close as possible to the upper left corner (coordinates (0, 1)), representing 100% sensitivity and 100% specificity. A completely

Table 1. Comparative performance of 31 machine learning methods

Group	Method	tp	fn	fp	tn	MCC	accuracy rate (%)
HMM	HMM	50	3	5	48	0.850	92.45
Bayes	CNB	49	4	3	50	0.868	93.40
	NaiveBayes	48	5	3	50	0.850	92.45
	NaiveBayesSimple	48	5	3	50	0.850	92.45
	NaiveBayesUpdateable	48	5	3	50	0.850	92.45
	AODE	50	3	7	46	0.814	90.56
Neural Net	MultiplayerPerceptron	49	4	3	50	0.968	93.40
	SMO	49	4	4	49	0.849	92.45
	RBFNetwork	48	5	6	47	0.793	89.62
	Logistic	45	8	5	48	0.756	87.73
	VotedPerceptron	46	7	10	43	0.680	83.96
Meta	LogitBoost	47	6	5	48	0.793	89.62
	MultiBoostAB	47	6	7	46	0.755	87.73
	MultiClassClassifier	45	8	5	48	0.756	87.73
	ThresholdSelector	44	9	5	48	0.738	86.79
	ADABoost	46	7	8	45	0.717	85.84
Trees	NBTree	47	6	5	48	0.793	89.62
	LMT	47	6	6	47	0.774	88.67
	ADTree	47	6	8	45	0.736	86.79
	J48	45	8	12	41	0.624	81.13
	ID3	44	7	14	37	0.594	76.41
Lazy	LBR	48	5	3	50	0.850	92.45
	IB1	49	4	15	38	0.656	82.07
	Kstar	48	5	14	39	0.651	82.07
	IBk	49	4	16	37	0.639	81.13
	LWL	41	12	14	39	0.510	75.47
Rules	PART	44	9	11	42	0.623	81.13
	DecisionTable	41	12	11	42	0.566	78.30
	Ridor	41	12	11	42	0.566	78.30
	JRip	42	11	13	40	0.548	77.35
	NNge	31	22	2	51	0.591	77.35

random guess would give a point along a diagonal line (the so-called line of no-discrimination) from coordinates (0, 0) to (1, 1) the left bottom to the top right corners.

Figure 1 shows the ROC graph for the classifiers evaluated in this work. Since the performance of all classifiers were above the no-discrimination line, this figure shows only the upper left quadrant of the graph. This is done only for comparison purposes, so as to amplify the differences between classifiers. The top classifiers are identified in the ROC space: HMM (Hidden Markov Model [2]), SMO (Sequential Minimal Optimization algorithm for support vector machine [10]), MLP (Multilayer Perceptron neural network [15]), LBR (Lazy Bayesian Rules classifier [15]), CNB (Complement class Naive Bayes [11]), and Bayes. This last one comprehends three different versions: Naive Bayes, NaiveBayesSimple and

4 Discussion and Conclusions

Several methods for detecting promoters in nucleotide sequences were compared. Table 1 shows that, considering only the accuracy rate, HMM and the Bayesian methods (including LBR and excluding AODE) were those that performed better. The results of another methods (table 2) are approximately similar to those obtained in this work. Also, they obtained the best results with neural network-based approaches, a fact that was confirmed in our work. Recall that no effort was done to fine-tune parameters in our experiments. Results were obtained using the default parameters, except for HMM where the threshold was set empirically. Therefore, it could be reasonable to expect slight better performances.

The ROC graph shows the difference between methods more clearly than table 1. Possibly, the small differences in performance of the methods tested may be due to the small number of instances of the data set. A single misclassified instance may lead to an error of almost 2%.

In general, the probabilistic methods, including HMM and Naive Bayes classifiers achieved better results than other classifiers. Possibly, this is due to the specific nature of biological sequences, where uncertainty is also present. That is why Bayesian methods have been traditional in bioinformatics, for tasks such as sequence alignment, pattern recognition and others. Similarly, some neural network-based methods, namely SMO and MLP, obtained results competitive to those of Bayesian methods, possibly due to its ability to establish complex hiperplanes in the problem space. This shows its adequacy for dealing with biological sequences. Regarding MCC, again a Bayesian and a neural-network method, respectively, CNB and MLP, were those that performed better.

Traditional methods for induction of decision-trees, such as ID3 and C45, are considered baseline in many data mining applications. However, for the data set and standard parameters used in this work they did not performed so well as others. This was confirmed by analyzing the MCC of those classifiers.

Future work include the use of ensemble methods for associating classifiers and the application of these methods to the detection of promoters in DNA sequences of eukariotes, as part of an automatic gene detection system.

References

1. Baldi, P., Brunak, S., Chauvin, Y., Andersen, C.A.F., Nielsen, H.: Assessing the Accuracy of Prediction Algorithms for Classification: an Overview. *Bioinformatics* 16, 412–424 (2000)
2. Durbin, R., Eddy, S., Krogh, A., Mitchison, G.: *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge (1998)
3. Ephraim, Y., Merhav, N.: Hidden Markov Processes. *IEEE T. Inform. Theory* 48, 1518–1569 (2002)
4. Fawcett, T.: An Introduction to ROC Analysis. *Pattern Recogn. Lett.* 27, 861–874 (2006)
5. Harley, C., McClure, W.: Compilation and Analysis of Escherichia coli Promoter DNA Sequences. *Nucleic Acids Res.* 11, 2237–2255 (1983)

6. Kohavi, R.: A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection. In: 14th Int. Joint Conf. on Artificial Intelligence, pp. 1137–1143 (1995)
7. Nelson, D.L., Cox, M.M.: *Lehninger Principles of Biochemistry*, 4th edn. W.H. Freeman, Chicago (2006)
8. Matthews, B.W.: Comparison of the Predicted and Observed Secondary Structure of T4 Phage Lysozyme. *Biochim. Biophys. Acta* 405, 442–451 (1975)
9. Mount, D.W.: *Bioinformatics: Sequence and Genome Analysis*. CSHL Press, Woodbury (2001)
10. Platt, J.: Fast Training of Support Vector Machines Using Sequential Minimal Optimization. In: Schoelkopf, B., Burges, C., Smola, A. (eds.) *Advances in Kernel Methods - Support Vector Learning*. MIT Press, Cambridge (1998)
11. Rennie, J., Shih, L., Teevan, J., Karger, D.: Tackling the Poor Assumptions of Naive Bayes Text Classifiers. In: Fawcett, T., Mishra, N. (eds.) 20th Int. Conf. on Machine Learning, pp. 616–623. AAAI Press, Menlo Park (2003)
12. Sing, T., Sander, O., Beerenwinke, N., Lengauer, T.: ROCr: Visualizing Classifier Performance in R. *Bioinformatics* 21, 3940–3941 (2005)
13. Towell, G., Shavlik, J., Noordewier, M.: Refinement of Approximate Domain Theories by Knowledge-based Artificial Neural Networks. In: 8th National Conference on Artificial Intelligence, pp. 861–866. AAAI Press, Menlo Park (1990)
14. Weinert, W., Lopes, H.S.: Neural Networks for Protein Classification. *Appl. Bioinformatics* 3, 41–48 (2004)
15. Witten, I.H., Frank, E.: *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd edn. Morgan Kaufmann, San Francisco (2005)
16. Wolfsberg, T., McEntyre, J., Schuler, G.: Guide to the Draft Human Genome. *Nature* 409, 824–826 (2001)

A Holistic View of Evolutionary Rates in Paralogous and Orthologous Genes

Deng Pan and Liqing Zhang

Department of Computer Science, Virginia Tech., USA
{dengpan, lqzhang}@vt.edu

Abstract. Much work has been done on the evolutionary divergence of duplicated genes in protein coding regions. However, this remains a partial view, and little is known how duplicated genes diverge in non-coding regions, which can play an important role in the evolutionary preservation of duplicated genes. Here we compared the evolutionary rates of different parts of duplicated genes in the human and mouse genomes, including 5'-UTRs, coding regions, 3'-UTRs, and 500 bps upstream and downstream of the untranslated regions. Results show that compared to orthologous genes, the ratios of the genetic distance of non-coding regions such as UTRs and upstream and downstream regions vs. synonymous substitutions tend to be smaller in paralogous genes, suggesting that synonymous substitutions benefit more greatly from relaxed selective constraints than noncoding regions. Moreover, we also examined the most frequent types of rate comparison among different genic regions in both orthologs and paralogs and found that the ranks of the most frequent types of rate comparison differ little between human paralogs and mouse paralogs, but differ greatly from those in orthologs, suggesting that duplication enables changes of evolutionary dynamics along different parts of genes. We also classified duplicated genes into three categories based on their chromosomal locations, tandem duplicates, intra-chromosomal duplicates, and inter-chromosomal duplicates. Results show that in both the human and mouse, selective constraint (measured by K_a/K_s) on intra-chromosomal duplicates tends to be much lower than that on tandem duplicates, which in turn is significantly lower than that on inter-chromosomal duplicates. This shows that genomic location is an important factor in determining the evolutionary divergence of duplicated genes.

1 Introduction

Gene duplication is generally believed to be one of the most efficient ways to generate genetic novelty. Right after gene duplication, it has been predicted and also empirically shown that the evolutionary rate should increase due to the subsequent relaxation of selective constraints [1]. Occasionally, the rate increase could also be caused by the action of positive selection that diversified the duplicates (e.g. [2]). Many studies have been devoted to the evolutionary divergence

of coding regions of duplicated genes to address the questions such as how frequent adaptive evolution is after duplication, and how duplicated genes diverge and gets preserved after duplication. The most recent work started to examine whether genes are duplicated equally and some progress has been made on the asymmetry of rates of evolution due to the inequality of birth of duplicated genes (e.g. [3]).

However, all the current studies have been focused on the evolutionary divergence of duplicated genes in their protein coding regions, the evolutionary divergence of noncoding regions has been largely ignored. In order to address this issue, we compared the evolutionary divergence of human and mouse paralogous genes in their noncoding regions including 5' UTRs, 3' UTRs, and 500 bps noncoding upstream and downstream of the untranslated regions with that of coding regions. To put this into context, we used orthologous genes between human and mouse as reference. We also examined the location effect on the evolutionary divergence of duplicated genes. Previous studies suggest that location is an important factor in determining the evolutionary fate of duplicated genes [3]. Taking advantage of the current dataset, we tested the location effect by partitioning the paralogous gene pairs into three categories, tandem duplicates where two paralogous genes on the same chromosome are separated from each other by no more than one non-homologous gene, intra-chromosomal duplicates where two paralogous genes are on the same chromosome but are not tandem duplicates, and inter-chromosomal duplicates where two paralogous genes are on different chromosomes. We compared the distribution of K_a/K_s in these three types of paralogs to examine whether selective constraints differ among them.

2 Methods

Selection of paralogous and orthologous gene pairs. We collected all the 5'UTR, 3'UTR, Coding sequences (CDSs) and peptide sequences of the human and mouse protein coding genes from the ENSEMBL version 46. We used the longest transcript of a gene if the gene has multiple alternative splicing forms. We also retrieved Ensembl's gene family tables of these two species. We included singleton families, i.e. families have only one member, in our dataset. We removed genes shorter than 50 amino acids in both our sequence collection and the family tables. Then, we selected the families that have no more than 5 members for analyses to reduce the amount of dependence in pairwise comparisons. We obtained the paralogous gene pairs by pairing genes within each family and the orthologous gene pairs by pairing genes of difference species that share the same ENSEMBL family id. Since we intended to compare 5'UTR, 3'UTR and CDS regions of these gene pairs simultaneously, we required that each gene should have all these three parts. At the same time, we excluded the genes whose 5'UTR are shorter than 20 bps and 3'UTR are shorter than 40 bps [4].

Sequence alignment. We did the pairwise alignment for these gene pairs using the GAP program [5], which uses a global optimal alignment algorithm. The parameters for the GAP program are the same as in [4]: the mismatch

penalty is -3 for DNA and the PAM120 scoring matrix for peptide alignments; the match is 10; the gap open penalty is 50; the gap extension penalty is 5; and the longest penalized gap is 10. CDSs were aligned by using the corresponding peptide sequence alignments as templates. To ensure the homology of a gene pair, we further removed the gene pairs that have an aligned overlapping percentage, i.e. the proportion of the length of the overlapping region to that of the total alignment, less than 50% in their CDS alignments. Consequently, we got 2,772 and 2,296 paralogous gene pairs in human and mouse respectively, and 11,394 orthologous gene pairs between human and mouse.

Substitution Rates. We estimated the evolutionary rates of 5'UTR, 3'UTR and CDS between the members of homologous gene pairs. The rates were estimated in terms of the number of substitutions per bp under the Kimura's two-parameter model [6]. For CDSs, we also estimated the synonymous (K_s) and nonsynonymous (K_n) rates using the YN00 method implemented in the PAML software package [7]. We excluded all gaps when calculating the rates.

3 Results and Discussion

We obtained 2,772 and 2,296 paralogous gene pairs in human and mouse respectively, and 11,394 orthologous gene pairs between human and mouse. All gene pairs contain UTRs (untranslated regions) and CDS (coding sequences). We also examined the 500 bps immediately upstream and downstream of the untranslated regions. These regions should be enriched with regulatory elements and their sequence evolution can shed light on the divergence of gene expression patterns [8]. The larger number of orthologs than paralogs is because for orthologs, singleton genes can be used to be matched between species, but for paralogs, only those families that have at least two members can generate paralogous gene pairs.

UTR divergence in paralogs and orthologs. The sequence identities are similar between the paralogs of human and mouse, but smaller than that of the orthologs in both 5'UTRs and 3'UTRs. 90% of the sequence identities in 5'UTR are between 37.7% and 72.7% in human paralogs, between 37.8% and 91.3% in mouse paralogs, while between 42.6% and 89% in human-mouse orthologs. In 3'UTR, 90% of them are between 39.7% and 77.5% in human paralogs, between 40.2% and 85.0% in mouse paralogs, while between 44.2% and 85.9% in human-mouse orthologs.

Up/down-stream divergence in paralogs and orthologs. The pattern of sequence identities is similar to that of UTR regions. The sequence identities of up and down-stream regions are similar between the paralogs of human and mouse, but smaller than those of the orthologs. The up-stream region in orthologues (55%) has a slightly higher average identity than the down-stream region (48%). 90% of the sequence identities in up-stream regions are between 36.7% and 57% in human paralogs, between 37.5% and 62.4% in mouse paralogs, while between 38.7% and 78.1% in human-mouse orthologs. In down-stream regions, 90% of them are between 37.1% and 66.6% in human paralogs, between 37.7%

and 67.2% in mouse paralogs, while between 38.5% and 68.8% in human-mouse orthologs.

CDS divergence in paralogs and orthologs. The sequence identities of CDSs in either DNA level or protein level tend to be higher than those of UTRs. 90% of the sequence identities in CDSs are between 42.8% and 86.2% in human paralogs, between 42.8% and 86.5% in mouse paralogs, while between 55.6% and 92.2% in human-mouse orthologs. Similar to UTRs, the results show more conservative CDSs alignments in orthologs than in paralogs, but the degree of conservation is not as great as that of UTRs.

Relative substitution rates within genes. To see how different parts of a gene evolve comparatively, we also obtained the relative substitution rate between different parts of the genes. To facilitate the comparison, we used the K_s of the coding region as the common denominator, and the distances of all other regions as the numerator. For each pair of genes (orthologs or paralogs), the divergence time is the same for different regions of the pair, therefore, the ratio between certain K distance vs. K_s represents the ratio of evolutionary rates in the corresponding regions. The distribution of the relative rates is shown in Figure 1. All rates of UTRs vs. the K_s of CDSs have a peak at around 0.2 when compared with the K_s of CDSs (Figure 1A-D). The distribution of the distance ratios of 5'UTR vs. K_s is similar to that of the distance ratios of 3'UTR vs. K_s for both human paralogs and mouse paralogs (Figure 1B,D). In contrast, for the human and mouse orthologs, the distance ratios of 5'UTR vs. K_s (or 3'UTR vs. K_s) tend to be much higher than those for both human and mouse paralogs, suggesting that gene duplication might confer a greater degree of relaxed selective constraints on synonymous substitutions than on substitutions in untranslated regions, which lead to a higher rate increase in synonymous substitutions than substitutions in untranslated regions. It has to be mentioned that increasing evidence suggests that synonymous mutations are not neutral [9]. As there seems little reason for possible slowdown of rates of evolution in untranslated regions in duplicated genes, the current observation offers yet another supporting evidence for the non-neutral evolution of synonymous mutations.

Similar to the patterns observed in UTRs, the distance ratios of upstream (or downstream) regions vs. K_s in paralogs also tend to be smaller than those in orthologs. It is expected that because of redundancy after gene duplication, there is relaxed selective constraints on mutations on every part of a gene, however, the magnitude of relaxation might differ in different regions. As evolution rates of noncoding regions such as UTRs and upstream and downstream regions of UTRs tend to be higher than coding regions under usual circumstances, relaxation of selective constraints on these regions might not have as a big effect as on coding regions. The K_a/K_s distribution is as what we expect, i.e., K_a/K_s tend to be higher in paralogs than in orthologs, which is the result of reduced purifying selections on nonsynonymous substitutions in coding regions.

As within a gene, the distance comparison is the same as the rate comparison, we thus sorted the distances in different regions of the same genes and counted the frequency of each type of rate comparison. Figure 2 shows

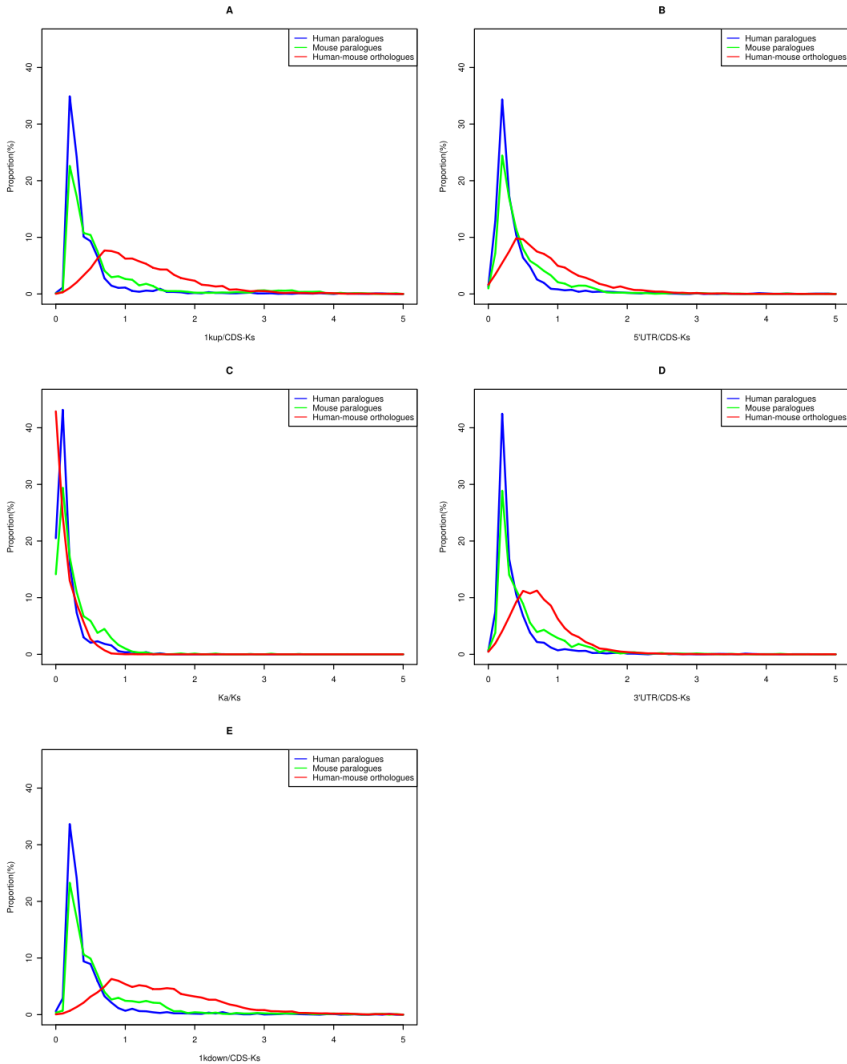


Fig. 1. Distributions of relative substitution rates for (A). K in 500 bp upstream region vs. K_s in CDS, (B). K in 5'UTR vs. K_s in CDS, (C). K_a/K_s in CDS, (D). K in 3'UTR vs. K_s in CDS, and (E). K in 500 bp downstream vs. K_s in CDS.

the top 25 most frequent types of rate comparisons using human paralogs as the reference. To examine the effect of duplication further, we divided the orthologs into three types, orthologs where both human and mouse genes are singletons (ortholog_{ss}), orthologs where either the human or mouse gene has duplicated copies (ortholog_{sd}), and orthologs where both human and mouse genes have paralogs (ortholog_{dd}). It is interesting to see that the most frequent

type of rate comparison for both paralogs and orthologs is “*CDS* < *5UTR* < *3UTR* < *5hup* < *5hdn*”, which means, the *K* distance is the highest in 500 bps downstream of 3’UTRs, followed by 500 bps upstream of 5’UTRs, 3’UTRs, 5’UTRs, and coding regions. The ranks of different rate comparisons are very similar between human paralogs and mouse paralogs. The ranks of different rate comparisons are also very similar within different types of orthologs. However, the ranks differ greatly between orthologs and paralogs. For example, the second most frequent type of rate comparison in human paralogs or mouse paralogs is “*CDS* < *5UTR* < *3UTR* < *5hdn* < *5hup*”, however, it only ranked sixth most frequent type of rate comparison in orthologs, and instead the second most frequent type of rate comparison in orthologs is “*CDS* < *3UTR* < *5UTR* < *5hup* < *5hdn*”. This suggests that relaxed selective constraints in duplicated genes lead to a larger magnitude of rate increases in 3’UTRs than in 5’UTRs. Therefore, clearly, the evolutionary dynamics in different parts of the genes differ depending on whether the genes arose from duplication or speciation. As numerous studies have shown that duplication leads to reduced selective constraints and provide opportunities for duplicated copies to explore what would have been impossible prior to duplication, the observation here further suggests that duplication can also alter the evolutionary rate comparisons in duplicated genes as compared to their orthologous counterparts.

Caution must be taken when interpreting the results. As non-coding regions tend to evolve faster than coding regions and there are no good models on the evolution of non-coding regions, it is extremely difficult to study these regions and make reliable sequence alignments. Although we take steps to ensure the quality of the sequence alignments, it is still possible that some of the alignments are of low quality. Moreover, for the non-coding regions that are fast-evolving, making reliable sequence alignments is nearly impossible. Therefore, the studied non-coding regions might be the ones that are more conserved. Nonetheless, as we are making a relative rate comparison across different genic regions and using the frequency of the rate comparison to extrapolate the relative abundance of each type, the results should be quite robust.

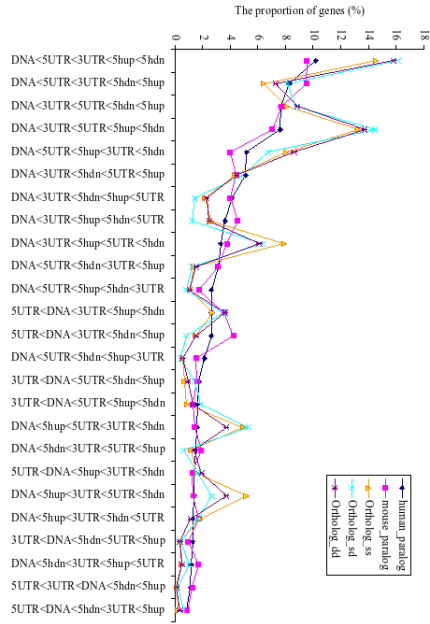


Fig. 2. The 20 most common types of rate comparisons along different regions of genes

Tandem duplicates, intra-chromosomal duplicates, vs. inter-chromosomal duplicates.

It has been observed that the difference in evolutionary rates between duplicated genes tends to be much greater for the duplications where one copy has been relocated (i.e. moved to a different genomic location, [3]). This suggests that location is an important factor in determining the evolutionary divergence and fate of duplicated genes. In order to examine how location has affected the sequence divergence of the paralogs in our data, we classified all the paralogs into three categories according to their current genomic locations, tandem duplicates where two paralogous genes have no more than one spacer (spacer refers to genes that are not homologous to the duplicates, see [10] for detailed definition), intra-chromosomal duplicates where two paralogous genes are on the same chromosome but are separated by more than one spacer), and inter-chromosomal duplicates where two paralogous genes are on different chromosomes. We examined the K_a/K_s distributions in the three types of duplicates to see whether they have any systematic difference in selective constraints.

Figures 3 and 4 show the cumulative distributions of K_a/K_s for the three types of duplicates in the human and mouse. There is a pronounced difference between the three distributions. In particular, the intra-chromosomal duplicates have a higher proportion of duplicates with large K_a/K_s than TAGs have, which in turn have higher proportion of duplicates with large K_a/K_s than inter-chromosomal duplicates. The three distributions in the mouse show a similar pattern to those in the human. All the observed differences are statistically significant (in human: inter-chromosomal vs. TAGs, $p - value < 2.2e - 16$; TAGs vs. intra-chromosomal, $p - value < 1.599e - 07$; in mouse: inter-chromosomal vs. TAGs, $p - value < 2.2e - 16$; TAGs vs. intra-chromosomal, $p - value < 3.273e - 07$). Therefore, our observation offers

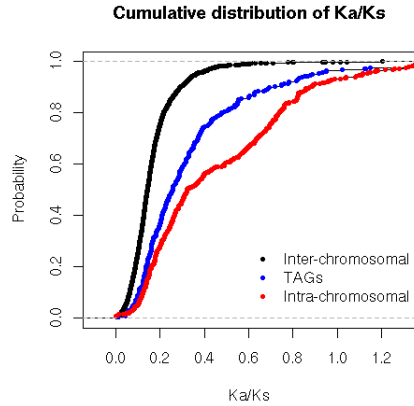


Fig. 3. The cumulative distribution of K_a/K_s for different types of duplicates in human

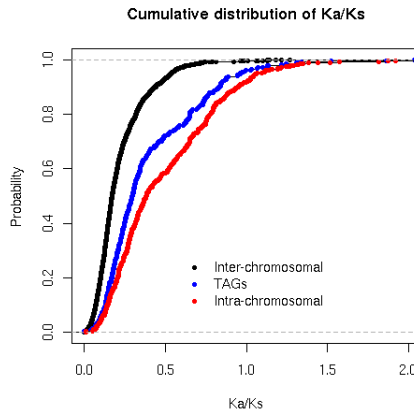


Fig. 4. The cumulative distribution of K_a/K_s for different types of duplicates in mouse

additional evidence that relative location of duplicated genes can play an important role in affecting the subsequent divergence of gene copies.

So why does tandem duplicates and intra-chromosomal duplicates tend to have higher K_a/K_s values than inter-chromosomal duplicates? It is likely that genes on the same chromosome tend to have more constant interactions such as gene conversions than duplicated copies that are on different chromosomes. Although gene conversion should not affect synonymous and nonsynonymous mutations differently, the known GC biased mismatch repair mechanism does introduce a bias, which could contribute to a higher number of mutations. Because there are more nonsynonymous sites in a gene than synonymous sites, we expect that the bias GC mismatch repair will introduce more nonsynonymous mutations than synonymous mutations, leading to an increase in K_a/K_s . However, although this seems to be able to explain at least in part the higher K_a/K_s values in tandem duplicates and intra-chromosomal duplicates than inter-chromosomal duplicates, it has some difficulty in explaining why tandem duplicates have lower K_a/K_s than intra-chromosomal duplicates. As the frequency of gene conversion decreases with the increase of chromosomal distances between two duplicated genes on the same chromosome, it is likely that intra-chromosomal duplicates tend to have lower rates of gene conversion than tandem duplicates. So if gene conversion can cause an increase in K_a/K_s , then our current observation that tandem duplicates tend to have lower K_a/K_s than intra-chromosomal duplicates is not expected. Therefore, there might be other reasons for the difference. Future studies can be directed to identify the mechanisms of duplication and features of duplication, which might help address the question.

References

1. Ohno, S.: Evolution by gene duplication. Springer, New York (1970)
2. Conant, G.C., Wagner, A.: Asymmetric sequence divergence of duplicate genes. *Genome. Res.* 13(9), 2052–2058 (2003)
3. Cusack, B.P., Wolfe, K.H.: Not born equal: increased rate asymmetry in relocated and retrotransposed rodent gene duplicates. *Mol. Biol. Evol.* 24(3), 679–686 (2007)
4. Makalowski, W., Boguski, M.S.: Evolutionary parameters of the transcribed mammalian genome: An analysis of 2,820 orthologous rodent and human sequences. *Proc. Natl. Acad. Sci. USA* 95(16), 9407–9412 (1998)
5. Huang, X., Chao, K.M.: A generalized global alignment algorithm. *Bioinformatics* 19(2), 228–233 (2003)
6. Kimura, M.: A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *J. Mol. Evol.* 16(2), 111–120 (1980)
7. Yang, Z.: PAML: A program package for phylogenetic analysis by maximum likelihood. *Comput. Appl. Biosci.* 13(5), 555–556 (1997)
8. Guo, Y., Jamison, D.C.: The distribution of SNPs in human gene regulatory regions. *BMC Genomics* 6, 140 (2005)
9. Hirsh, A.E., Fraser, H.B., Wall, D.P.: Adjusting for selection on synonymous sites in estimates of evolutionary distance. *Mol. Biol. Evol.* 22(1), 174–177 (2005)
10. Shoja, V., Zhang, L.: A roadmap of tandemly arrayed genes in the genomes of human, mouse, and rat. *Mol. Biol. Evol.* 23(11), 2134–2141 (2006)

An SVM-Based Algorithm for Classifying Promoter-Associated CpG Islands in the Human and Mouse Genomes

Leng Han^{1,2}, Ruolin Yang², Bing Su², and Zhongming Zhao^{1,*}

¹ Department of Psychiatry and Center for the Study of Biological Complexity, Virginia Commonwealth University, Richmond, VA 23298, USA

² State Key Laboratory of Genetic Resources and Evolution, Kunming Institute of Zoology, Chinese Academy of Sciences, Kunming, Yunnan 650223, China
zzhao@vcu.edu

Abstract. CpG islands (CGIs) are clusters of CpG dinucleotides in GC-rich regions and represent an important gene feature of mammalian genomes. Several algorithms have been developed to identify CGIs. Here we applied Support Vector Machine (SVM), a machine learning approach, to classify CGIs that are associated with the promoter regions of genes. We demonstrated that our SVM-based algorithm had much higher sensitivity and specificity in classifying promoter-associated CGIs than other algorithms, and had high reliability. The advantages of SVM in our method and future improvements were discussed.

Keywords: CpG islands (CGIs); Support Vector Machine (SVM); promoter; human; mouse.

1 Introduction

1.1 General Features of CpG Islands (CGIs)

CGIs are clusters of CpG dinucleotides in GC-rich regions, usually ~1 kb long [1]. They are identified in the promoter regions of approximately 50% of genes in vertebrate genomes and are considered gene markers. The number of CGIs varies greatly among mammalian genomes. However, this great difference is unlikely attributed to the CGIs located in the promoter regions because the numbers of promoter-associated CGIs in the two genomes are very close [2]. These CGIs tend to present stronger CGI characteristics than the CGIs in other genomic regions [3]. Therefore, it is important to distinguish and classify the promoter-associated CGIs from the others. So far, this issue has not been successfully resolved.

1.2 Existing Algorithms for Identifying CGIs

Several algorithms have been reported in the literatures [1, 4, 5], but all these algorithms utilize sequence properties only and fail to identify promoter-associated CGIs

* Corresponding author.

specifically. At present, there is one algorithm, which was named CpGProD and was developed by Ponger and Mouchiroud [6], to classify promoter-associated CGIs by predicting the potential promoters and their orientation. The sensitivity and specificity for CGI predictions was 0.38 and 0.62, respectively, when promoter-associated CGIs on human chromosome 22 were classified, using the threshold 0.3. For the whole human genome, the sensitivity and specificity were 0.27 and 0.60, respectively. Overall, the performance of their predictions is not strong. Promoter-associated CGIs are longer, have larger observed over expected CpG ratios ($\text{Obs}_{\text{CpG}}/\text{Exp}_{\text{CpG}}$) and higher G+C content than other CGIs. Such differences form theoretical bases for distinguishing promoter-associated CGIs from other CGIs in mammalian genomes.

1.3 Support Vector Machine (SVM)

SVM is a popular and powerful machine learning approach for classifications. The details of SVM are available in Vapnik et al. [7]. The basic principle of SVM is based on a training set of n samples, $\{x_i, y_i\}$, $i=1, \dots, n$, where x_i are vectors with features, $y_i \in \{+1, -1\}$ and denote the classes. The classifier can be generated by the following formula:

$$f(x) = \text{sgn} \left\{ \sum_{i=1}^n \alpha_i y_i K(x_i, x) + b \right\} \quad (1)$$

where K is the kernel function that defines the feature space, α_i and b are parameters that are optimized during training process in order to maximize margin separating hyper-plane from among the many that can separate the positive from negative samples. SVM has been broadly applied in pattern recognition [8, 9].

1.4 Application of SVM in CGI Classification

Recent studies have demonstrated the utility of SVM in prediction of methylation status of CpGs [10] or CGIs [11]. To date, there is no algorithm that applies SVM in classification of promoter-associated CGIs. In this study, we developed an SVM-based method to distinguish and classify promoter-associate CGIs.

2 Materials and Methods

2.1 Genome Sequences and Gene Annotation

We downloaded the reference sequences of human and mouse genomes and their gene annotations from the National Center for the Biotechnology Information (NCBI) (<ftp://ftp.ncbi.nlm.nih.gov/genomes/>). We used the Perl script developed by Jiang and Zhao [12] to extract the location of transcriptional start sites (TSS) of each gene. The promoter region of each gene was marked using the definition in Ioshikhes and Zhang [3], i.e., -500 to +1500 bp around its TSS.

2.2 CGIs Identified by Existing Algorithms

We identified CGIs in genomic sequences using Takai and Jones' algorithm, which is optimized for searching CGIs associated with the 5' end of genes in the human and other mammalian genomes. The search criteria were: GC content $\geq 55\%$, $\text{Obs}_{\text{CpG}}/\text{Exp}_{\text{CpG}} \geq 0.65$, and length ≥ 500 bp. To evaluate the performance of CGI classification by CpGProD, we also identified CGIs according to the criteria in CpGProD: GC content $\geq 50\%$, $\text{Obs}_{\text{CpG}}/\text{Exp}_{\text{CpG}} \geq 0.60$, and length ≥ 500 bp.

We extracted the location and parameters of each CGI and then mapped it to the corresponding chromosome. A CGI was classified as a promoter-associated CGI if it overlapped with a promoter region; all others were classified as a non-promoter-associated CGI.

2.3 Classification of Promoter-Associated CGIs by SVM

CGIs identified by Takai and Jones' algorithm and their characteristics (length, GC content and $\text{Obs}_{\text{CpG}}/\text{Exp}_{\text{CpG}}$ ratio) were used to generate SVM classifiers. There were three steps in the procedure (Figure 1). (1) We generated a training set of sub-samples that were randomly selected from all the available CGIs in a genome according to the ratio of promoter-associated CGIs over non-promoter-associated CGIs. We empirically set the sub-sample size 4000 because such a size is expected to have sufficient statistical power and to minimize the probability of retaking a sample (i.e., a CGI) from the pool (e.g., $\sim 37,000$ human CGIs). For each sub-sample set, a classifier was generated based on the features of two CGI groups. The process was repeated 50 times, thus, 50 independent classifiers were generated. (2) To examine whether promoter-associated CGIs are distinguishable from the others by our trained SVM classifiers, we performed a 5-fold cross-validation (Figure 1). (3) For each classifier generated by the training set, we then classified CGIs in a test set (sample size 4000), which was randomly selected from the whole CGI pool. Similarly, we repeated this step 50 times to assess the performance. In each round, we calculated sensitivity (SN), specificity (SP), accuracy (AC) and correlation coefficient (CC). The equations (2-5) are below:

$$SN = TP / (TP + FN) \quad (2)$$

$$SP = TN / (TN + FP) \quad (3)$$

$$AC = (TP + TN) / (TP + TN + FP + FN) \quad (4)$$

$$CC = \frac{(TP \times TN) - (FN \times FP)}{\sqrt{(TP + FN) \times (TN + FP) \times (TP + FP) \times (TN + FN)}} \quad (5)$$

where TP are true positives (predicted promoter-associated CGIs and true promoter-associated CGIs); TN are true negatives (predicted non-promoter-associated CGIs and true non-promoter-associated CGIs); FP are false positives (predicted

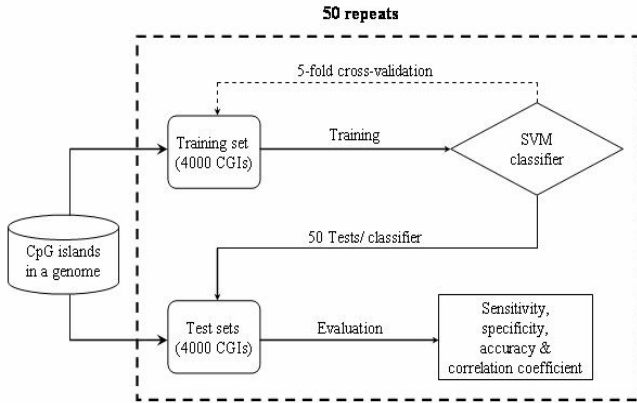


Fig. 1. Flow chart of SVM-based algorithm for CGI classification

promoter-associated CGIs but false promoter-associated CGIs) and FN are false negatives (predicted non-promoter-associated CGIs but false non-promoter-associated CGIs). There were a total of 2500 rounds in the computational procedure.

The SVM-based algorithm was implemented by Python PyML package (<http://pyml.sourceforge.net/>).

3 Results

3.1 General Features of CGIs in the Human and Mouse Genomes

Consistent with previous studies [3], in both the human and mouse genomes, promoter-associated CGIs presented stronger CGI characteristics than non-promoter-associated CGIs, but the variation of the three parameters within each class of CGIs were even stronger (Figure 2). This made it difficult to distinguish promoter-associated CGIs.

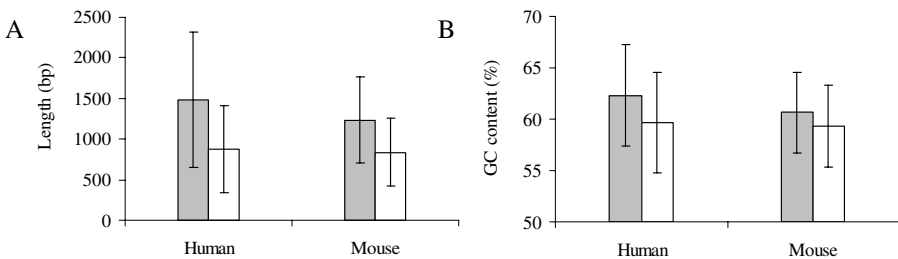


Fig. 2. Characteristics of CGIs in the human and mouse genomes. Gray bars: promoter-associated CGIs; white bars: non-promoter-associated CGIs. (A) Length (bp). (B) GC content (%). The Obs_{CpG}/Exp_{CpG} shows the similar pattern.

3.2 Evaluation of SVM-Based Algorithm in Classifying CGIs

We calculated sensitivity (SN), specificity (SP), accuracy (AC) and correlation coefficient (CC) to assess how effective the SVM was in classifying the promoter-associated CGIs. The SN, SP, AC and CC values were, respectively, 0.83, 0.67, 0.75 and 0.48 in the human genome. The corresponding values in the mouse genome were 0.77, 0.64, 0.71 and 0.42 (Figure 3). Importantly, sensitivity and specificity in our study were much higher than the CpGProd [10].

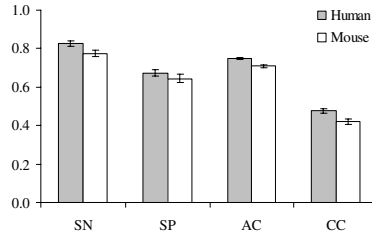


Fig. 3. Performance of CGI classifications in the human and mouse genomes based on SVM algorithm. SN: sensitivity; SP: specificity; AC: accuracy; and CC: correlation coefficient.

3.3 Reliability of SVM-Based Algorithm in Classifying CGIs

As shown in Figure 3, small standard errors of SN, SP, AC and CC of the 2500 sets of sub-samples indicated a high reliability of our method. We further examined the distributions of SN, SP, AC and CC for reliability and found a very narrow distribution range in the human genome (Figure 4). The result was similar in the mouse genome (data not shown).

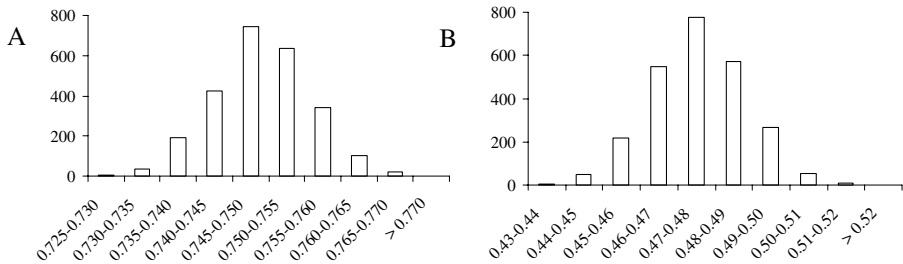


Fig. 4. Frequency (number of counts, Y-axis) of test results for random samples in the human genome: (A) Accuracy, (B) Correlation coefficient. Note that the scale on X-axis varies in each sub-figure. The sensitivity and specificity show the similar patterns.

4 Discussion

4.1 Comparison of Classified Results with Previous Study

Promoter-associated CGIs can not easily be distinguished from non-promoter-associated CGIs individually, which makes their classification challenging. The performance of our

method is better than that of the CpGProD. This is likely because of the following points. (1) Large-scale data is now available. The predictive classifier in CpGProD was generated according to the features of only 818 CGIs in the human genome and 163 CGIs in the mouse. Our prediction was based on all the genomic CGIs and well annotated gene data, especially human genes. (2) The SVM in our method is a powerful tool for classifications. (3) The initial CGI data used in our method were obtained by Takai and Jones' algorithm, which was empirically refined to identify gene-associated CGIs. This might help the classification of CGIs. However, this seems not the case. We used the same parameters in the CpGProD to identify CGIs on human chromosome 22, and then used our SVM-based algorithm to classify CGIs. The sensitivity and specificity were 0.85 and 0.78, respectively, which again were much higher than the corresponding values by the CpGProD.

4.2 Future Improvement

Our algorithm could classify 71% (mouse) ~75% (human) of CGIs correctly. We may improve our algorithm by the following ways. First, in the present study, we used only 3 parameters in SVM to identify the CGIs. In particular, GC content is significantly correlated with $\text{Obs}_{\text{CpG}}/\text{Exp}_{\text{CpG}}$ ($r=0.14$, $P=6.6\times 10^{-168}$) in the human CpG island sequences. SVM should be more powerful to process the samples when more features are employed. Therefore, more features, such as representative motifs in the promoter regions, methylation status of CpGs, and location of CGIs may be added to improve our algorithm. Second, we may increase sample size, number of tests and rounds of cross-validations in identification. Third, we may improve the performance of classifiers by using alternative Kernel functions rather than linearity.

Acknowledgments. We thank Dr. Amiee Potter for critically reading the manuscript. This project was supported by the Thomas F. and Kate Miller Jeffress Memorial Trust Fund and a NARSAD Young Investigator Award to Z. Zhao.

References

1. Lander, E.S., et al.: Initial Sequencing and Analysis of the Human Genome. *Nature* 409, 860–921 (2001)
2. Jiang, C., Han, L., Su, B., Li, W.H., Zhao, Z.: Features and Trend of Loss of Promoter-associated CpG Islands in the Human and Mouse Genomes. *Mol. Biol. Evol.* 24, 1991–2000 (2007)
3. Ioshikhes, I.P., Zhang, M.Q.: Large-scale Human Promoter Mapping Using CpG Islands. *Nat. Genet.* 26, 61–63 (2000)
4. Gardiner-Garden, M., Frommer, M.: CpG Islands in Vertebrate Genomes. *J. Mol. Biol.* 196, 261–282 (1987)
5. Takai, D., Jones, P.A.: Comprehensive Analysis of CpG Islands in Human Chromosomes 21 and 22. *Proc. Natl. Acad. Sci. U.S.A.* 99, 3740–3745 (2002)

6. Ponger, L., Mouchiroud, D.: CpGProD: Identifying CpG Islands Associated with Transcription Start Sites in Large Genomic Mammalian Sequences. *Bioinformatics* 18, 631–633 (2002)
7. Vapnik, V.N.: *Statistical Learning Theory*. Wiley, New York (1998)
8. Brown, M.P., et al.: Knowledge-based Analysis of Microarray Gene Expression Data by Using Support Vector Machines. *Proc. Natl. Acad. Sci. U.S.A.* 97, 262–267 (2000)
9. Ramaswamy, S., et al.: Multiclass Cancer Diagnosis Using Tumor Gene Expression Signatures. *Proc. Natl. Acad. Sci. U.S.A.* 98, 15149–15154 (2001)
10. Bhasin, M., Zhang, H., Reinherz, E.L., Reche, P.A.: Prediction of Methylated CpGs in DNA Sequences Using a Support Vector Machine. *FEBS Lett.* 579, 4302–4308 (2005)
11. Fang, F., Fan, S., Zhang, X., Zhang, M.Q.: Predicting Methylation Status of CpG Islands in the Human Brain. *Bioinformatics* 22, 2204–2209 (2006)
12. Jiang, C., Zhao, Z.: Mutational Spectrum in the Recent Human Genome Inferred by Single Nucleotide Polymorphisms. *Genomics* 88, 527–534 (2006)

CDGMiner: A New Tool for the Identification of Disease Genes by Text Mining and Functional Similarity Analysis

Fang Yuan¹ and Yanhong Zhou¹

¹ Hubei Bioinformatics and Molecular Imaging Key Laboratory, Huazhong University of Science and Technology, Wuhan 430074, China
yhzhou@hust.edu.cn

Abstract. In the post-genomic era, the identification of genes involved in human disease is one of the most important tasks. Disease phenotypes provide a window into the gene function. Several approaches to identify disease related genes based on function annotations have been presented in recent years. Most of them, starting from the function annotations of known genes associated with diseases, however, can not be used to identify genes for diseases without any known pathogenic genes or related function annotations. We have built a new system, CDGMiner, to predict genes associated with these diseases which lack detailed function annotations. CDGMiner is implemented mainly by two phases, text mining and functional similarity analysis. The performance of CDGMiner was tested with a set of 1506 genes involved in 1147 disease phenotypes derived from the OMIM database. Our results show that, on average, the target gene was in the top 13.60%, and the target gene was in the top 5% with a 40.70% chance. CDGMiner shows promising performance compared to other existing tools.

1 Introduction

Human diseases are commonly associated with both the environmental factors and genetic factors. Identifying genes that are likely to be involved in human genetic disease is of vital importance for both understanding disease pathogenic mechanism and improving clinical practice [1,2]. Through complex-trait linkage studies, many disease genes are located within one or more specific chromosomal regions, which regularly contains hundreds of genes [3]. Clearly, it is a tedious, labor-intensive task to perform mutation analysis for the large number of genes one by one in the regions of interest. Therefore, prioritizing candidate genes by computational algorithms is prevalent to speed the identification of disease genes.

During recent years, some bioinformatics application tools for prioritizing candidate genes have been developed and released to public [4-15]. They can be classified into two broad, while not mutually exclusive categories. One is based on single type of genomic data (functional annotations, sequences, expression profiles et al.) [4-12], the other is based on the integration of various types of genomic data [13-15]. The later strategy is a recent development and a trend for the future. However, Franke, et al. [15] observed that GO annotation is the most effective data resource, and the accuracy based on GO was slightly improved by adding other types of

data. Therefore, improving the accuracy of function annotation based methods will greatly help improve the accuracy of multi-information fusion methods.

Most function annotation based tools often suffer from annotation bias as they can not deal with diseases lacking known causative genes. Neither can these tools handle known genes lacking sufficiently detailed function annotations. We have developed a computational web tool, CDGMiner, to prioritize genes on a chromosomal region for the diseases lacking known causative genes. Starting with the assumption that genes involved in phenotypically similar diseases share similarity in their functional annotation[16], CDGMiner makes prediction by using a combination of text mining and gene-function similarity analysis. The process of the text mining of CDGMiner adopts similar approach as that of G2D, and the function-similarity analysis uses a novel algorithm based on the GO directed acyclic graphs (DAG). Specifically, we explored how to make use of the structure of DAG to analyze function similarity for improving the performance of identifying disease genes. G2D, POCUS, and other methods are also based on GO annotation. However, these methods just find candidates with the same functions (shared GO terms). Different from these methods, we search for candidate genes of similar functions with similar but not necessarily shared GO terms.

2 Methods

2.1 The Flowchart of CDGMiner

Given a phenotypical definition of a disease, and a chromosomal region where to search for causative genes associated with the disease of interest, the flowchart of CDGMiner for prioritizing candidate genes is shown in Fig.1.

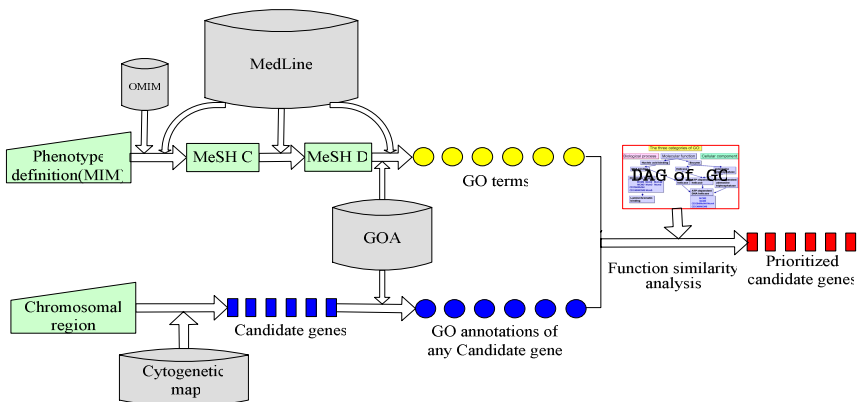


Fig. 1. The flowchart of CDGMiner

It includes the following two phases:

1. mining a set of GO terms related to the given disease from MEDLINE/PubMed [17] and protein function annotation database GOA provided by the GO Project[18], and weighting them. The text mining and weighting procedures are described in detail in section 2.2.

- Prioritizing the candidate genes in the chromosomal region of interest by function similarity analysis. Firstly, we retrieve the GO terms of the candidate genes in the given chromosomal region. Secondly, based on the GO DAG, the function similarities between the GO terms of candidate genes and the set of GO terms mined from biomedical literatures are analyzed, and the possible relations of candidate genes to the disease are computed. Finally, we prioritize the candidates according to the relation degrees of the candidates to the disease. The algorithm for function similarity analysis is presented in detail in section 2.3.

2.2 Mining Disease-Related GO Terms from the Biomedical Literature Database

The OMIM IDs are taken from OMIM-morbid map lists[19] as the phenotypical definition of a disease. For a given OMIM ID, there are some MEDLINE references, given as PubMed IDs annotations in OMIM database. And most of the references in MEDLINE / PubMed are annotated with a controlled set of MeSH C terms ('Disease Category', describing the phenotype of the disease) [20].

However, there are often not enough literature in MEDLINE to directly relate disease phenotypes, represented by MeSH C terms, to gene product functions, represented by GO terms. While genes can be related to diseases symptoms through chemical features of molecules, they are represented by MeSH D terms ('Chemicals & Drugs'). Mesh D terms are used as an alternative way to annotate the biomedical literature in MEDLINE / PubMed, similar to MeSH C terms. In these cases, we take MeSH D terms as the intermediate to increase the relation between MeSH C terms and GO terms.

The GO annotation database by the GO projects provides comprehensive, non-redundant functional annotations of gene products, i.e. proteins. And every annotation submitted to GO in the database must be attributed to a source, which is a kind of evidence to support the association between the gene product and the GO term. The evidence is often a PubMed ID linked to a biomedical reference in MEDLINE. Therefore, we can use the biomedical literature to associate MeSH D terms with GO terms.

Obviously, according to the transitive relations of MeSH C to OMIM, MeSH D to MeSH C, and MeSH D to GO terms, we can retrieve a set of GO terms associated with the disease phenotype defined by the OMIM ID, and weight them by the following algorithms.

The Evaluation of the Association between MeSH C and OMIM

Given a disease phenotype as o , and the set of MeSH C terms associated with o as $MC:\{mc_1, mc_2, \dots, mc_m\}$, the relation degree of any MeSH C term mc_i in MC to o is scored by:

$$mc2o(o, mc_i) = \frac{|mc_i|}{|o|} \quad (1)$$

where $|o|$ represents the number of the MEDLINE references linked by o , and $|mc_i|$ represents the number of the ones annotated by mc_i .

The Evaluation of the Association between MeSH D and MeSH C

To obtain the association between MeSH C (mc) and MeSH D (md), the co-occurrence frequency of mc and md is considered. We assume that these two terms are highly related if they frequently appear together. Here the strength of the association between mc and md is evaluated by counting the co-occurrences of both terms.

$$md2mc(mc, md) = \frac{|mc \cap md|}{|mc \cup md|} \quad (2)$$

where $|mc \cap md|$ denotes the number of the MEDLINE references annotated by both mc and md , and $|mc \cup md|$ denotes the number of the MEDLINE references annotated by either mc or md .

The Evaluation of the Association between GO Term and MeSH D

Given a MeSH D term as md , and a GO term as go , the association between md and go is assigned a score by the following formula:

$$go2md(md, go) = \frac{|md|}{|go|} \quad (3)$$

Where $|go|$ represents the number of the MEDLINE references which have been found to be the evidences of go in the GOA database, and $|md|$ represents the number of the ones annotated by md among the $|go|$ references.

The Evaluation of the Association between GO Term and OMIM

According to the transitive relations of MeSH C to OMIM, MeSH D to MeSH C, and MeSH D to GO term, we weight the contribution of a GO term go to the disease phenotype o by the following formula:

$$go2o(o, go) = mc2o(o, mc) \times md2mc(mc, md) \times go2md(md, go) \quad (4)$$

Here, the higher scored GO terms are considered to contribute more to the disease phenotype.

2.3 Prioritizing the Candidate Genes by Function Similarity Analysis

Based on the set of the weighted GO terms mined from MEDLINE, we can evaluate the possibility for a gene to be the cause of a disease. The function similarity of any given two terms is the basis to make the evaluation.

The Evaluation of the Function Similarity of Two GO terms

Recent years, the GO consortium has established a set of structured, controlled vocabularies to describe gene functions. The GO ontologies are structured as directed acyclic graphs (DAG), which are similar to hierarchies but differ in that a child (more specialized) GO term has more than one parent (less specialized) GO terms. The DAG structure provides a rich description about gene functions.

Every GO term in DAG must obey the true path rule: if the child term describes the gene product, all its parent terms must also apply to that gene product. This means that two GO terms will be considered to have similar functions, if they have one or more same ancestors. Obviously, through their co-ancestry nodes, the two terms can be connected by one or more, short or long paths. Here, we define that the unit distance between two adjacent GO terms is 1, and of course the distance of a GO term to itself is 0. Under the definition, the functional similarity degree of two terms, go_i and go_j , is calculated according to the shortest distance between them, as shown in the following equation.

$$go2go(go_i, go_j) = \frac{1}{sp(go_i, go_j) + 1} \quad (5)$$

Where $sp(go_i, go_j)$ denotes the distance of the shortest path between go_i and go_j .

The Evaluation of the Relation Degree of Genes to Disease

Here, given the set of GO terms mined as $S_G: \{go_1, go_2, \dots, go_m\}$, and the set of GO terms related to a candidate gene g in the target region as $S_C: \{go_1, go_2, \dots, go_n\}$.

Before defining the relation between a candidate gene and the disease, we evaluate the relation degree of a single GO term go_k of the candidate to the disease phenotype o , as the following formula.

$$cgo2o(o, go_k) = go2o(o, go_h) \times \max\{go2go(go_i, go_k) \mid go_i \in S_G\} \quad (6)$$

Where go_h is the one in S_G most similar to go_k and makes $go2go(go_h, go_k)$ the max value.

Finally, the relation of the candidate gene g to the disease phenotype is assigned a score by the following equation.

$$g2o(o, g) = \frac{1}{m} \sum_{i=1}^n cgo2o(o, go_i) \quad (7)$$

The genes with high scores are considered as the candidate genes related to the given disease with high possibility.

3 Test and Comparison

3.1 Test and Result

The performance of CDGMiner is tested with a set derived from Morbid Map table in the OMIM database. To assure the validity of the test, the diseases (OMIM phenotypes) in the test set must be assumed to have no known related genes, and must contain at least one related GO terms by mining MEDLINE literature via PubMed. The items meted the conditions in Morbid Map table were all selected into the test set, in a completely fair manner. The set (dataset1) consisted of 1506 genes involved in 1147 OMIM phenotypes. Here, we picked a fixed target gene for each disease phenotype, and took a region of 30 Mb around this gene as the target region. On average each region of interest contained 337 genes.

To evaluate the performance from the test results, we calculated sensitivity(S_n) and specificity(S_p). Sensitivity refers to the frequency (% of all prioritizations) of target genes that are ranked above a particular threshold position. Specificity refers to the percentage of genes ranked below this threshold. To allow comparison between data sources we plotted the receiver operating characteristic (ROC) curves, from which sensitivity/specificity values can be easily deduced. The area under this curve (AUC) is a standard measurement of the performance of this algorithm.

For the test set, CDGMiner reached a high performance, as the dash-dotted curve shown in Fig.2. The result shows that 40.70% target genes were ranked among the top 5% of the prioritized list, and on average the target gene was within the top 13.60%. The AUC score was as high as up to 86.40%, indicating that CDGMiner could identify correct disease gene with high effectiveness.

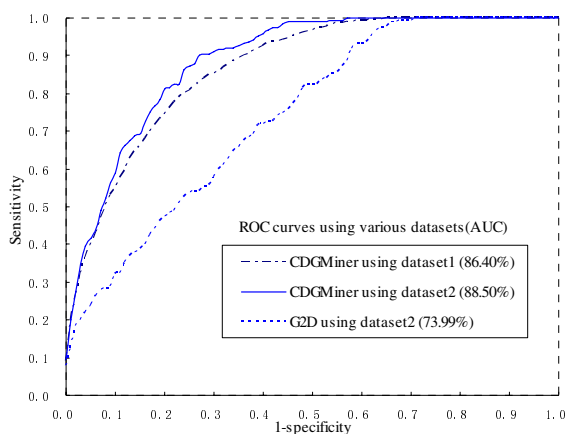


Fig. 2. The ROC curves of CDGMiner and G2D. The dash-dotted curve for CDGMiner was the result using the dataset1 which was all the data available from OMIM, the solid curve for CDGMiner and the dotted curve for G2D were the results using the dataset2 which was derived from the website of G2D(http://www.ogic.ca/projects/g2d_2/table_benchmark_complex.html).

3.2 Comparison with Other Similar Methods

Like CDGMiner, the Phenotype method of G2D can also make predictions for such diseases lacking detailed GO terms. G2D identifies disease genes through sequence similarity analysis based on the assumption that sequence homology indicates common functions.

To compare CDGMiner to the phenotype method of G2D in a fair manner, the same test dataset(dataset2), derived from the website of G2D has been used. The corresponding ROC graphs of CDGMiner and G2D were plotted as the solid curve and the dotted curve, respectively (Fig.2). the AUC for CDGMiner is 88.50%, higher than that for G2D (73.99%). Besides, the ROC of CDGMiner was also completely higher than that of the later method at every point. These results suggest that CDGMiner performs better than G2D. The difference between CDGMiner and the phenotype method of G2D is that the former identifies disease genes by functional

similarity analysis, while the later does this by sequence similarity analysis. Our results suggest that identifying disease genes through functional similarity analysis is likely more reliable than through sequence similarity analysis.

4 Conclusions

We have presented a novel web tool, CDGMiner, to prioritize candidate disease genes located in a region of interest by a combination of text mining and function similarity analysis. Based on disease-related GO terms mined from biomedical literature, CDGMiner is able to retrieve not only candidates with the same GO annotations but also those with similar GO annotations. The search quality of CDGMiner can be improved with increasing accuracy and completeness of GO DAG and gene function annotations by GO terms.

Unlike many existing tools for diseases with rich function annotations, CDGMiner makes predictions mainly for diseases lacking known causative genes and/or diseases with known genes but lacking sufficient function annotations. However, the GO terms mined from biomedical literatures may not be accurately related to a disease and this may result in predictions with poor precision. For diseases that lack detailed function annotations, it might be a better idea to combine our method which is based on mined GO terms with other methods that are based on sequence, expression profiles, and so on. Following this idea, we would be able to improve the performance of our method in the future.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (Grants Nos. 90608020, 30370354 and 90203011), NCET-060651, the National Platform Project of China (Grant No. 2005DKA64001), and the Ministry of Education of China (Grants Nos. 20050487037 and 505010).

References

1. Lander, E.S., Linton, L.M., Birren, B.: Initial Sequencing and Analysis of the Human Genome. *Nature* 409, 860–921 (2001)
2. Venter, J.C., Adams, M.D., Myers, E.W.: The Sequence of the Human Genome. *Science* 291, 1304–1351 (2001)
3. McCarthy, M.I., Smedley, D., Hide, W.: New Methods for Finding Disease-Susceptibility Genes: Impact and Potential. *Genome Biology* 4, 119 (2003)
4. Driel, M.A.V., Cuelenaere, K., Kemmeren, P.P.C.W., Leunissen, J.A.M., Brunner, H.G., Vriend, G.: GeneSeeker: Extraction and Integration of Human Disease-related Information from Web-based Genetic Databases. *Nucleic Acids Res.* 33, W758–W761 (2003)
5. Masseroli, M., Galati, O., Pincioli, F.: GFINDER: Genetic Disease and Phenotype Location Statistic Analysis and Mining of Dynamically Annotated Gene Lists. *Nucleic Acids Res.* 33, W717–W723 (2005)
6. Tiffin, N., Kelso, J.F., Powell, A.R.: Integration of Text and Data-mining Using Ontologies Successfully Select Disease Gene Candidates. *Nucleic Acids Res.* 33, 1544–1552 (2005)

7. Perez-Iratxeta, C., Bork, P.: Update of the G2D Tool for Prioritization of Gene Candidates to Inherited Diseases. *Nucleic Acids Res.* 35(Web Server issue), W212–W216 (2007)
8. Zhang, P., Zhang, J., Sheng, H.: Gene Functional Similarity Search Tool (GFSST). *BMC Bioinformatics* 7, 135 (2006)
9. Lopez-Bigas, N., Ouzounis, C.A.: Genome-wide Identification of Genes Likely to be Involved in Human Genetic Disease. *Nucleic Acids Res.* 32, 3108–3114 (2004)
10. Adie, E.A., Adams, R.R., Evans, K.L., Porteous, D.J., Pickard, B.S.: Speeding Disease Gene Discovery by Sequence Based Candidate Prioritization. *BMC Bioinformatics* 6, 55 (2005)
11. Oti, M.: Predicting Disease Genes Using Protein–protein Interactions. *J. Med. Genet.* 43, 691–698 (2006)
12. Jianzhen, X., Yongjin, L.: Discovering Disease-genes by Topological Features in Human Protein-protein Interaction Network. *Bioinformatics* 22, 2800–2805 (2006)
13. Adie, E.A., Adams, R.R., Evans, K.L., Porteous, D.J., Pickard, B.S.: SUSPECTS: Enabling Fast and Effective Prioritization of Positional Candidates. *Bioinformatics* 22, 773–777 (2006)
14. Aerts, S.: Gene Prioritization through Genomic Data Fusion. *Nat. Biotechnol.* 24, 537–544 (2006)
15. Franke, L., Bakel, H., Fokkens, L., Jong, E.D., Egmont-Petersen, M., Wijmenga, C.: Reconstruction of a Functional Human Gene Network, with an Application for Prioritizing Positional Candidate Genes. *Am. J. Hum. Genet.* 78, 1011–1025 (2006)
16. Jimenez-Sanchez, G., Barton, C., David, V.: Human Disease Genes. *Nature* 409, 853–855 (2001)
17. MEDLINE/PubMed, <http://www.ncbi.nlm.nih.gov/PubMed>
18. EBI GOA project, <http://www.ebi.ac.uk/GOA/index.html>
19. OMIM, <http://www.ncbi.nlm.nih.gov/sites/entrez?db=OMIM>
20. MeSH, <http://www.ncbi.nlm.nih.gov/MeSH>

Designing Genetic Regulatory Network with Fuzzy Logic

Jianwei Shen^{1,2}, Yi Wang², and Zengrong Liu²

¹ Department of Mathematics, Xuchang University, Xuchang 461000, China
jwshen8@yahoo.com.cn

² School of Science and Institute of System Biology, Shanghai University,
Shanghai 200444, China

Abstract. Based on the fact that fuzzy system is universal approximator, we made use of fuzzy system to design the genetic regulatory network. First, we viewed the differential equation as difference equation and classified the input data in term of time point and obtained the output data, so the system can be viewed as the two-input and one output system. Second, we fuzzy the spaces of input and output. Third, according to the theory of fuzzy system, we designed a fuzzy genetic regulatory network. In this paper, we established connection between discrete data and continuous data, and investigated the dynamical behaviors of gene network.

1 Introduction

Fuzzy logic is based on fuzzy set theory and especially on the concept of a fuzzy set. Informally, a fuzzy set is a set with imprecise boundaries, in which the transition from membership to non-membership is gradual rather than abrupt. A fuzzy set F in a universe of discourse U is characterized by a membership function μ_F , which associates each element $u \in U$ with a grade of membership $\mu_F(u) \in [0, 1]$. Note that a classical set A in U is a special case of a fuzzy set with all membership values $\mu_F(u) \in \{0, 1\}$. A fuzzy implication is viewed as describing a fuzzy relation between the fuzzy sets forming the implication. A fuzzy rule, such as “if x is A then y is B ” is implemented by a fuzzy implication which has a membership function $\mu_{A \rightarrow B}(x, y) \in [0, 1]$. Note that $\mu_{A \rightarrow B}(x, y)$ measures the degree of truth of the implication relation between x and y . A set of related fuzzy rules forms a fuzzy rule base that can be used to infer fuzzy results in the form of fuzzy sets.

Fuzzy logic offers an appealing method for describing phenomena by a set of rules and data sets. These data sets relate directly to concepts used on a daily basis, such as “fast”, “strong” or “high”, while the rules express knowledge approximately the same way a human expert would. An example of a fuzzy rule would be “if the car is fast, then the force applied to the brakes is strong”.

Given these characteristics, fuzzy rules are easy to understand, since they are very similar to the way a person might express knowledge. This also makes them

attractive for us in domains where experts are available and can seed the systems with a number of effective rules from the outset.

From above, we can know that it is very effective to deal with data by using fuzzy set theory. At present, Microarray technologies have produced tremendous amounts of gene expression data. In this paper, we can design the genetic network by using fuzzy set and describe the dynamical behavior of gene.

2 Designing the Fuzzy Genetic Regulatory Network

Generally, a genetic network can be expressed by a set of nonlinear differential equations with each gene expression level as variables

$$\frac{dx(t)}{dt} = f(x(t)), \tag{1}$$

where $x(t) = (x_1(t), \dots, x_n(t))^T \in R^n$ and $f = (f_1, \dots, f_n)^T : R^n \mapsto R^n$, $x_1(t)$ is the expression level (mRNA concentrations) of gene i at time t . Assume that there are total t time points for a given experimental condition from microarray, i.e, t_1, \dots, t_m , f_1 is a C^1 class nonlinear function

Generally, we can obtain mRNA concentrations of gene $x_i(t)$ at time t , but do not know the expression of $f(x(t))$, i.e., we do not know the interaction relation between gene $x_i(t)$. To study the genetic regulatory network, we should obtain the the expression of $f(x(t))$, according to the microarray data. In fact, it is impossible to find the exact $f(x(t))$. From the theory of fuzzy system, we know that fuzzy system is universal approximator [1]. So we will construct a fuzzy system according to the microarray data in different time points, and make the fuzzy system universal approximator of $f(x(t))$.

The equation (1) can be viewed as follows

$$\dot{x}_i(t_j) = \frac{x_i(t_{j+1}) - x_i(t_j)}{t_{j+1} - t_j} = f(x_i) = y_i(t_j), \tag{2}$$

For the microarray data, input data is $x(t) = (x_1(t), \dots, x_n(t))$, the output data is $y = (f(x_1), \dots, f(x_n))$, we can obtain $f(x_i)$ from (2)

Microarrays are based on the idea that, in every cell, at every states, only a fraction of the total DNA is transcribed into mRNA, which is subsequently translated into protein, the translated gene is said to be expressed. In general, Describing the expression level of gene is imprecise. To study the dynamical behavior of genes precisely, we design the fuzzy rules according to the gene expression level and deduce relation between genes.

It is assumed that each feature may be associated with several fuzzy sets, described in Table 1. This is different from the situation one would face in a Boolean gene network where the gene could be either expressed (binary value 1) or not (binary value 0), Obviously, since fuzzy logic is being used, membership can be different from 0 to 1 in any set at any given time.

The expression space of each feature is divided evenly, so that each fuzzy set has an equal support. The process is shown in Fig.1. There is overlapping among

adjacent fuzzy sets, so that more than one rule may be active for each expression value.

When we are using fuzzy logic, there is no need to normalize the expression data because each expression value will be turned into a membership value, which will be used in the rules. Linear transformations would be specially pointless, due to the fuzzy sets would be exactly the same, but dislocated in space. A nonlinear transformation of the data, such as the application of a log transform is possible, especially in the case where there may be a large variation in the expressions of the expression values where of the same order of magnitude.

Each fuzzy set must be associated with at least one rule.

Suppose we are given a set of desired input-output data from microarray data and equation (2) as follows

$$(x^{(1)}, y^{(1)}), \dots, (x^{(n)}, y^{(n)}), \tag{3}$$

In the following, we will describe the step to construct fuzzy system according the microarray data.

Step 1. Order the microarray data according to time points.

For the gene x_i , we order the gene according to time points as follows $x_i(t_1), \dots, x_i(t_n)$, where $t_1 < t_2 < \dots < t_n$

Step 2. Computing the y_i according to Eq.(2)

By using the Eq.(2), we can obtain the output data $y_i(t_j)$

Step 3. Divide the Input and Output Spaces into Fuzzy Regions

Assume that x_i is concentration of gene i , the domain internals of x_i and y_i are $[x_i^-, x_i^+]$, $[y_i^-, y_i^+]$, respectively, where domain interval of a variable means that most probably this variable will lie in this interval. Divide each domain interval into $2N + 1$ regions (N can be different for different variables and the lengths of these regions can be equal or unequal), denoted by $SN, \dots, S1$, CE , $B1, \dots, BN$, and assign each region a fuzzy membership function.

Step 4. Generate Fuzzy Rules from Given Data Pairs

Assume that $x_i^l(t_{j+1}), x_i^j(t_j); y_i^l(t_j)$ is input-output data. First, determine the degrees of input-output data in different regions. Section, assign a given input-output data to the region with maximum degree. Finally, obtain one rule from one pair of desired input-output data.

Step 5. Assign a Degree to Each Rule

Since there are usually lots of data pair and each data pair generates on rule, it is highly probable that there will be some conflicting rules, i.e., rules that have the same IF part but a different THEN part. One way to resolve this conflict is to assign a degree to each rule generated from data pairs, and accept only the rule from a conflict group that has maximum degree. In this way not only is the conflict problem resolved, but also the number of rules is greatly reduced.

We use the following product strategy to assign a degree to each rule: “ IF $x_i(t_{j+1})$ is A and $x_i(t_j)$ is B THEN $y_i(t_j)$ is C ”, the degree of this rule, denoted by $D(\text{rule})$, is defined as follows,

$$D(\text{rule}) = m_A(x_i(t_{j+1}))m_B(x_i(t_j))m_C(y_i(t_j)), \tag{4}$$

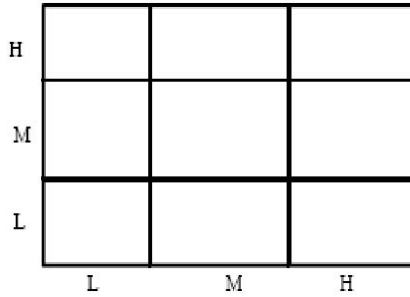


Fig. 1. The form of a fuzzy rule base

In fact, we often have some priori information about the data pair according to the experience of expert. So we design the degree of rule and should add the degree of these experiences of expert. If the degree of experiences is $m^{(1)}$, then

$$D(rule) = m_A(x_i(t_{j+1}))m_B(x_i(t_j))m_C(y_i(t_j))m^{(1)}, \tag{5}$$

Step 6. Create a Combined Fuzzy Rule Base.

Intuitively, $x_i(t_j)$ represents abscissa axis and $x_i(t_{j+1})$ represents axis of ordinates, we can select fuzzy set (for example, we select the three fuzzy set), see Fig.1 In Fig.1, L is fuzzy set which represents “Low expression Level”, M is fuzzy set which represents “Medium expression Level” and H is fuzzy set which represents “High expression Level”. We fill the boxes of the base with fuzzy rules according to the following strategy: a combined fuzzy rule base is assigned rules from either those generated from numerical data or linguistic rules. If there is more than one rule in one box of the fuzzy rule base, use the rule that has maximum degree. In this way, both numerical and linguistic information are codified into a common framework-the combined fuzzy rule base.

In Fig.2, abscissa axis represent concentration of gene and vertical axis represents membership function.

Step 7. Design the Fuzzy System Based on Fuzzy Rule Base.

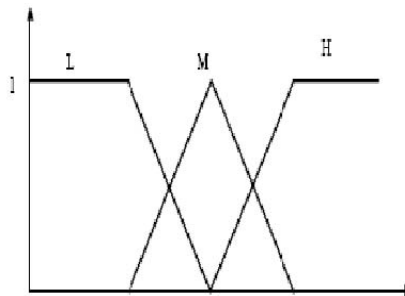


Fig. 2. Divisions of the input and output spaces into fuzzy regions

In this step, we use the following defuzzification strategy to determine the output control $y_i(t_j)$ for given input $(x_i(t_{t+1}), x_i(t_j))$

First, for input data $(x_i(t_{t+1}), x_i(t_j))$, we combine the antecedents of the fuzzy rule using product operation to determine the degree $m_{O^l}^l$, of the output control corresponding to input data, i.e $m_{O^l}^l = m_{I_1^i}(x_i(t_{j+1}))m_{I_2^i}(x_i(t_j))$

Where O^l denotes the output region of Rule l and I_i^i denotes the input region of Rule l for i th component. Thus, we use the centroid defuzzification formula to determine the output

$$y = \frac{\sum_{l=1}^K m_{O^l}^l \bar{y}^l}{\sum_{l=1}^K m_{O^l}^l}, \tag{6}$$

Where \bar{y}^l denotes the center value of region O^l (the center of a fuzzy region is defined as the point that has the smallest absolute value among all the points at which the membership function for this region has membership value equal to one), and K is the number of fuzzy rules in the combined fuzzy rule base.

From the Step 1 to Step 7, we can infer the fuzzy gene regulatory network as follows

$$\dot{x}_i(t_j) = \frac{\sum_{l=1}^K m_{O^l}^l \bar{y}^l}{\sum_{l=1}^K m_{O^l}^l}, \tag{7}$$

3 Fuzzy System Is Universal Approximate System of Original System

Lemma 1. Assume that $U \in R^n$ is a compact set, if $f(x)$ is real and continuous function, which defined in U for $\forall \epsilon > 0$. Then there exists a fuzzy system y which satisfy the following

$$\sup_{x \in U} |f(x) - y| < \epsilon, \tag{8}$$

From the reference [1], we can prove that (6) is universal approximator of $f(x)$ in arbitrary accuracy, so we can replace (1) with (7) and to study the genetic network.

4 Discussion and Conclusion

In this paper, we design the fuzzy genetic network represented by differential dynamical system, we design the fuzzy rules according to expressing level of gene, and fuzzy set theory. This method constructs a bridge between the continuous model and discrete data.

Acknowledgements. This work is partly supported by NSF of China (10672093), China Postdoctoral Science Foundation funded project (20070410717) and Natural Science Foundation of Henan Province.

References

1. Wang, L.X.: A Course in Fuzzy Systems and Control. Prentice Hall PTR, Englewood Cliffs (1996)
2. Wang, L.X.: Adaptive Fuzzy System and Control: Design and Stability Analysis. Prentice Hall, Englewood Cliffs (1994)
3. Ram, R., Chetty, M., Trevor, I.D.: Fuzzy Model for Gene Regulatory Network. 2006 IEEE Congress on Evolutionary Computation, pp. 1450–1455 (2006)
4. Wang, L.X., Mendel, J.M.: Fuzzy Basis Functions, Universal Approximation, and Orthogonal Least-Squares Learning. IEEE Trans. Neural. Network 3(5), 807–814 (1992)
5. Wang, L.X., Mendel, J.M.: Generating Fuzzy Rules by Learning from Examples. IEEE Trans. Syst. Man. Cyber. 22(6), 1414–1427 (1992)
6. Chen, L., Wang, R.: Designing Gene Regulatory Networks With Special Functions. IEEE Trans. Circuits Syst. I. 53(11), 2444–2450 (2006)
7. Chen, L., Aihara, K.: Stability of Genetic Regulatory Networks with Time Delay. IEEE Trans. Circuits Syst. I. 49(5), 602–608 (2002)

A Framework to Understand the Mechanism of Toxicity

Wenhui Huang and Liqing Zhang

Department of Computer Science, Virginia Tech, U.S.A
{wenhui, lqzhang}@vt.edu

Abstract. Computational identification of the mechanisms of drug toxicity is a very challenging problem. Little progress has been made thus far. In this work, we propose a novel framework to identify proteins involved in the chosen toxicities. Specifically, we used the proteins (bait proteins) that have been identified empirically to be involved in the toxicities to fish out additional proteins that might be strongly associated with the bait proteins in the protein-protein interaction network. We applied our method to 14 toxicities and manually validated two toxicities including bleeding disorders and urinary disorders. Literature research indicates that most of the newly identified proteins are involved in the toxicities in some degrees, and the networks identified are consistent with the known studies related to the toxicities.

1 Introduction

Developing a novel compound that can effectively bind to its target protein, known as drug's therapeutic target, has been a daunting task for the drug industry [1]. Moreover, the identification of drug's toxicity profile, especially the cases that happen in human therapy, become an impossible task due to the rare study on how compounds influence human biochemical pathways. Currently, there are several methods for screening the drug toxicity. One is the metabonomics method, which involves the usage of chemical analysis technique for evaluating the toxicity of drug-candidate compounds [2]. Although the approach is novel, the entire process is time and cost consuming and relies on the existence of biomarkers for evaluation. Several empirical methods exploit the genetic risk factors distributed among humans and use informative pharmacogenetic severe adverse drug reaction (SADR) studies to elucidate the link between genetic defects and specific SADR [3]. However, this requires the availability of large patient databases containing well-documented cases and carefully selected controls.

Other approaches include computational ligand screening methods, such as the quantitative structure activity relationship/structure activity relationship (QSAR/SAR)-based analysis and the statistical methods, which have been developed to emphasize the role of ligands or their components in the onset of toxicity. Unfortunately, none of the computational methods have been generally accepted and validated by real data. They are limited by factors such as the quality of molecular descriptors, the diversity of the compounds in the training set, and the efficiency of computing algorithm.

Recently, Lim et al. proposed a system approach to link a particular inherited phenotype symptom to a protein interaction network and suggested that many seemingly unrelated protein partners may work within one network in the pathogenesis [4]. It has been commonly accepted that many adverse drug events are induced by the interaction of drugs or their interactive metabolites with proteins that are named as the drug-induced toxicity-related proteins (DITRPs). By binding to these proteins, drugs or reactive metabolites disturb their normal physiological activities, which therefore directly or indirectly lead to toxicities. The toxicities may happen in ways of reversible inhibition/stimulation of therapeutic targets or some key proteins in other physiological pathways, blockage of drug transportation, disturbance of drug metabolism, encumbrance of drug excretion, excessive antigen-antibody interaction, and genetic susceptibility. Even more, the DITRPs may interweave into a network and thus further strengthen the possibility of the occurrences of specific drug toxicity [3-5]. Linking such DITRP networks to phenotypes of drug toxicities is important because it can shed light on the mechanisms of drug toxicities.

In this paper, we provide a method to build a DITRP network and an efficient algorithm to parse such network in order to find the most relevant unknown protein partners involved in the ontology of phenotypes of drug toxicities.

2 Methods

Datasets preparation. Two public databases are downloaded in Dec, 2007 and pre-processed in this study. One is the STRING protein-protein interaction database from the well-known ensemble database (<http://string.embl.de/>). This database contains 373 distinct organisms and more than 1.5 million proteins involved in the network. One key feature of STRING is that it provides an estimated score for every edge of protein interaction, which makes convenient to construct a weighted graph based on the score information. The other key feature of STRING is that it also provides the same protein across different species, which make the comparison between different species easier. The other database used in our study is Ditops (<http://bioinf.xmu.edu.cn/databases/ADR/>) that collects drug induced toxicity related proteins. It contains 618 distinct literature-reported DITRPs, 529 drugs/ligands, and 418 distinct toxicity terms. Those proteins are curated from many sources of publications and experimental data and thus have high possibility in association with specific symptom of drug toxicities.

The generation of bait protein sets. We use the Ditops data as bait protein sets and categorized data according to their phenotypes of toxicity. There are altogether 32 types of toxicities, but for our preliminary work, we randomly selected and examined 14 of the toxicities in the Ditops database (Table 1).

The generation of protein-protein interaction network. We downloaded the 65 Megabytes interaction data from the STRING database. As the focus of this paper is on humans, we only extracted human protein-protein interaction data and specifically choose the interaction edges that have weights larger than 500. In this way, we can

Table 1. Bait and protein numbers in each known Adverse Drug Reaction

Types of disorders	Bait protein number	Prey protein number
urinary system	20	12
Cardiovascular	6	5
Endocrine	9	5
Gastro-intestinal system	32	18
hearing and vestibular	8	3
heart rate and rhythm	6	10
Metabolic and nutritional	32	4
muscular-skeletal system	21	9
Neoplasms	5	2
Psychiatric	24	7
Red blood cell	6	2
Respiratory system	17	3
Platelet and bleeding	11	8
Liver	18	15

ascertain that the chosen proteins are on the strong bonding sub-network. A java program is implemented for the representation of graph data structure.

The Steiner-tree algorithm and its biological implications. Computational identification of proteins involved in certain drug toxicities is a challenging problem. There might be many proteins involved and complex interactions among them. However, we can take advantage of the proteins that have been empirically validated to be responsible for the toxicity (hereafter called bait proteins) and use them to “fish out” additional proteins (hereafter called prey proteins) that might also be involved in the toxicity. If we consider the interactions among these proteins in a network framework, we may convert this problem into a Steiner tree problem. The stein tree problem is to find the shortest path that connects the given vertices in a network. If we take the bait proteins as the Steiner points, we could identify the Steiner tree that includes prey proteins that minimizes the total length of the subnetwork. In this case, the prey proteins have practical meanings in that they have strong association with the bait proteins since they are in the same tree. Also, the newly identified prey proteins can be considered as “missing links” between the bait proteins. The shortest path also implies that the proteins in the tree have strongest interactions with one another. In this case, if bait proteins are associated with known drug-induced toxicity phenotype, the prey proteins are also highly likely to be ascribed to this phenotype of toxicity. Also, since Steiner trees always include the minimum set of vertexes that make the weigh of sub-graph minimum, in biological sense, it could be a useful tool in partitioning clusters within the protein-protein interaction network.

The most general form of Steiner-tree is: given a weighted graph $G(V, E, w)$ and a vertices subset $S \in V$, find a tree of minimal weight that includes all vertices in S . Though this problem is NP-complete [6], certain heuristic algorithms have been

designed to approximate the result within polynomial time [6]. Here, we implement the JAVA code of the algorithm stated in: <http://www.cs.mcgill.ca/~cphill1/SteinerTrees.html>

3 Results and Discussion

Figure 1 summarizes the computational steps that we used to identify proteins that might be involved in different toxicities. For our preliminary work, we analyzed altogether 14 toxicities from the Ditop database, shown in Table 2. The number of bait proteins that have been empirically identified to be associated with these toxicities

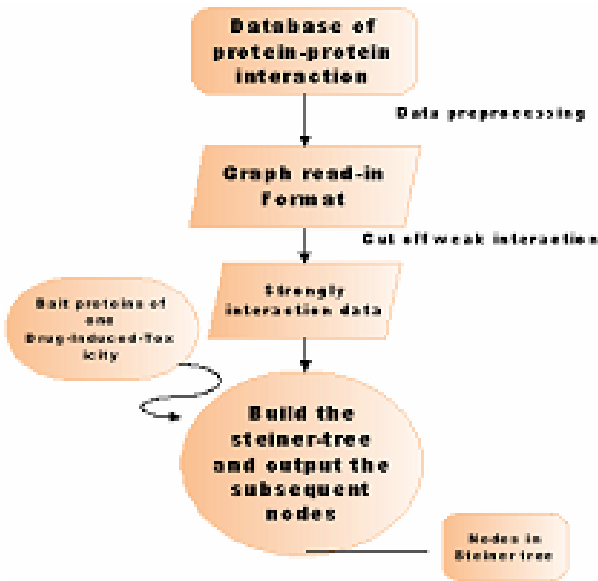


Fig. 1. The flowchart of the implementation of the whole program

ranges from 5 to 32. The number of prey proteins that are identified by our computational approach ranges from 2 to 18. It seems that the more bait proteins, the more prey proteins. The Gastro-intestinal system disorder involves the most number of proteins (=50) among the 15 toxicities, suggesting that the toxicity might be a result of many proteins affected in the network.

To evaluate the performance of our approach in identifying

the possible mechanisms of the toxicities, we manually examined the prey proteins that were identified in the bleeding and urinary toxicities. We searched the prey proteins in the literature to see whether they are involved in biochemical functions or pathways related to either blood or urinary disorders.

For the prey proteins identified in bleeding toxicity, we found that 10 out of the 11 prey proteins are to some degree related to bleeding disorder (table 2).

Close examination of the bleeding related proteins in the Steiner tree reveals some intriguing findings (Fig 2). There are two cycles and thus two putative pathways that closely overlap with each other, suggesting they may work together in the consequence of blood toxicity. Especially notable are the two pairs of proteins, Epoxide hydrolase 1 (EPHX1) and transcription factor GATA-4, which exist in one such putative pathway,

Table 2. Bait proteins that are associated with palate, bleeding and clotting disorders

Protein name	Pubmed evidence (Pubmed id)
Sodium/bile acid cotransporter	transcellular route and its permeation was partially affected by bile acid transporters, which lead to chemically modified heparin for long-term oral administration (17490773)
Aldo-keto reductase family 1 member B10	thyroid hormone (TH) induce Aldo-keto reductase genes that skin fibroblasts (15507505)
Transcription factor GATA-4	GATA-4 mediates 5-HT-induced growth of PASMCM (12615926)
ABCC2	No evidence found
TPMT Thiopurine S-methyltransferase	polymorphisms in TPMT gene may be responsible for approximately 12.5% of all leukopenia episodes in renal transplant recipients treated with azathioprine. Genotyping for the major TPMT variant alleles may be a valuable tool in preventing AZA toxicity and optimization of immunosuppressive therapy (16044099)
Cytochrome P450 2C9	clopidogrel is a CYP 2C9 inhibitor, is on on platelet inhibition by clopidogrel (17442686)
Erythropoietin precursor	Mechanism of increased mortality risk with erythropoietin treatment to higher hemoglobin targets (17942772)
SP1 Transcription factor	suppression of Sp1 expression and consequent down-regulation the downstream targets of Sp1 that are key to angiogenesis (17973266)
PTH Parathyroid hormone precursor	human platelets express the PTH1R. PTHrP can interact with this receptor to enhance human platelet activation induced by several agonists through a MAPK-dependent mechanism (17501718)
VEGF Vascular endothelial growth factor A	VEGF-A can stimulate platelet-derived growth factor (PDGF) receptors, which regulates MSC migration and proliferation, which a key therapeutic objective, since while increased neovascularization can be advantageous during tissue ischaemia, it is deleterious during tumourigenesis (17979880)
ABCC2	No evidence found

and thrombospondin and VEGF-A, which exist in the other putative pathway. This two pairs of proteins become the distinguished feature between two putative pathways. Zhu et al. [7] reported that the regulation of human microsomal EPHX1 expression by the transcription factor GATA-4, which when inhibited, may lead to hypercholanemia, suggesting that this putative pathway is controlled by EPHX1. For the other pathway, Straume et al. [8] reported that increased expression of VEGF-Receptors (FLT-1, KDR, NRP-1) and Thrombospondin-1 is associated with glomeruloid microvascular proliferation.

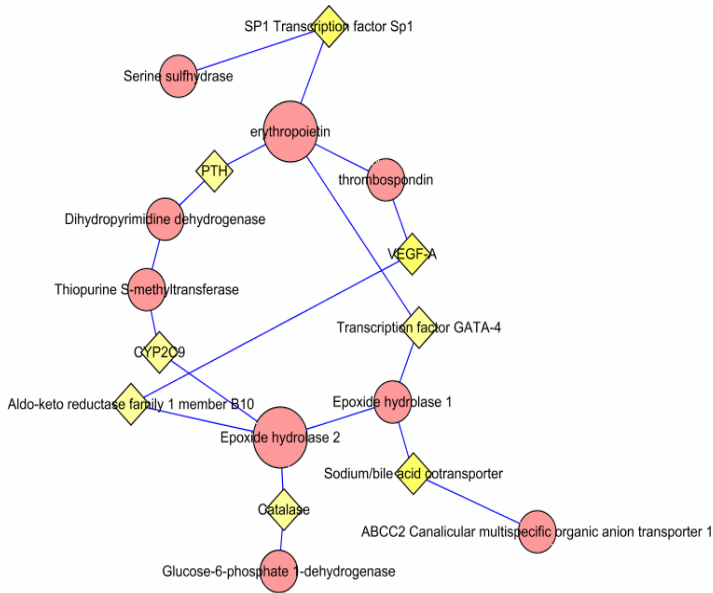


Fig. 2. The putative pathways identified by the Steiner-tree algorithm. The red circles represent the bait proteins that have been experimentally verified to be associated with Platelet and blood toxicity, and the yellow ones are prey proteins that are predicted by our algorithm.

We also found that the proteins associated with urinary toxicity are recovered by our algorithm. Fig. 3 shows the proteins identified by our approach. There are at least two cycles in the graph. If we take each cycle as a putative pathway, the two cycles reveal some interesting findings. For example, one of the two putative pathways (red connections) comprises 9 proteins with the nuclear receptor coactivator 1 (NRC1) being the key protein in the pathway. Two kinases and one phospholipase are also included in this putative pathway. It has been reported that coregulator phosphorylation can alter enzymatic activity and protein interaction [8]. Xiong et al. [9] further point out that progression of renal diseases is almost always associated with inflammatory processes and/or involve metabolic disorders of lipid and glucose, cell proliferation, hypertrophy, apoptosis, and hypertension, the importance of nuclear receptors and their coregulators in these contexts is thus very important.

In general, our preliminary work indicates that the Steiner-tree algorithm of searching for the subnetwork involved in various toxicities is very promising in generating the putative pathways which may lead to the drug induced toxicity. As we know, discovering what kind of pathways that foreign compounds may accidentally hit into is the very first step to elucidate such complicated mechanisms involved in toxicities. For the two diseases that we manually validated using the literature, most of the newly predicted proteins are associated with the corresponding diseases and furthermore, these proteins together with the bait proteins seem to form naturally some putative pathways that reveal their biochemical function in the corresponding toxicity

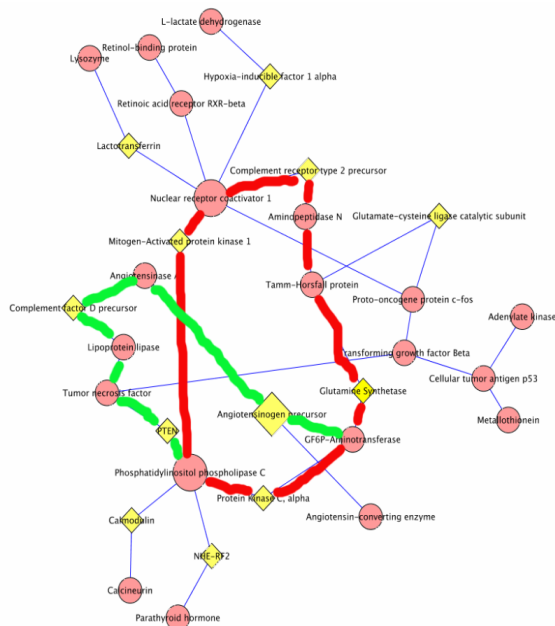


Fig. 3. The putative pathways built by the Steiner-tree algorithm, notice that the red circles represent the bait proteins which are experimentally verified to be associated with urinary toxicity, and the yellow ones are prey proteins which are predicted by our algorithm. There are two putative pathways, represented by red (pathway A) and green connections (pathway B).

phenotypes. Our current and future research includes several directions. One is to combine genome-wide gene expression data and the protein-protein interaction network data. Although our preliminary work shows very promising results, there is still limitation with protein-protein interaction data. For example, there might be many false positive interactions. Using expression data to validate the interactions will be a good way to alleviate this problem. Another improvement is to incorporate the knowledge in drug target development for those toxicities that have little empirical studies on bait proteins, either to expand the number of putative bait proteins or to predict bait proteins *in silico* so that we are not limited to only toxicities with much empirical studies.

References

1. Butcher, E.C.: Can Cell Systems Biology Rescue Drug Discovery? *Nat. Rev. Drug Discov.* 4(6), 461–467 (2005)
2. Nicholson, J.K., Connelly, J., Lindon, J.C., Holmes, E.: Metabonomics: A Platform for Studying Drug Toxicity and Gene Function. *Nat. Rev. Drug Discov.* 1(2), 153–161 (2002)
3. Wilke, R.A., Lin, D.W., Roden, D.M., Watkins, P.B., Flockhart, D., Zineh, I., Giacomini, K.M., Krauss, R.M.: Identifying Genetic Risk Factors for Serious Adverse Drug Reactions: Current Progress and Challenges. *Nat. Rev. Drug Discov.* 6(11), 904–916 (2007)

4. Lim, J., Hao, T., Shaw, C., Patel, A.J., Szabo, G., Rual, J.F., Fisk, C.J., Li, N., Smolyar, A., Hill, D.E., et al.: A Protein-protein Interaction Network for Human Inherited Ataxias and Disorders of Purkinje Cell Degeneration. *Cell* 125(4), 801–814 (2006)
5. Pujana, M.A., Han, J.D., Starita, L.M., Stevens, K.N., Tewari, M., Ahn, J.S., Rennert, G., Moreno, V., Kirchhoff, T., Gold, B., et al.: Network Modeling Links Breast Cancer Susceptibility and Centrosome Dysfunction. *Nat. Genet.* 39(11), 1338–1349 (2007)
6. Hwang, F., Richards, D., Winter, P.: *The Steiner Tree Problem*. North-Holland, Amsterdam (1992)
7. Zhu, Q.S., Qian, B., Levy, D.: Regulation of Human Microsomal Epoxide Hydrolase Gene (EPHX1) Expression by the Transcription Factor GATA-4. *Biochim. Biophys. Acta* 1676(3), 251–260 (2004)
8. Eikesdal, H.P., Dahl, O., Straume, O., Akslen, L.A.: Tumor Angiogenesis and Therapy Directed at the Neovasculature. *Tidsskr. Nor. Laegeforen* 124(15), 1919–1922 (2004)
9. Xiong, W., Tang, C.Q., Zhou, G.X., Chao, L., Chao, J.: In Vivo Catabolism of Human Kallikrein-binding Protein and Its Complex with Tissue Kallikrein. *J. Lab. Clin. Med.* 119(5), 514–521 (1992)
10. Wolf, G., Thaiss, F., Scherberich, J.E., Schoeppe, W., Stahl, R.A.: Glomerular Angio-tensinase A in the Rat: Increase of Enzyme Activity Following Renal Ablation. *Kidney Int.* 38(5), 862–868 (1990)
11. Papakonstanti, E.A., Stourmaras, C.: Association of PI-3 Kinase with PAK1 Leads to Actin Phosphorylation and Cytoskeletal Reorganization. *Mol. Biol. Cell* 13(8), 2946–2962 (2002)

FindSUMO: A PSSM-Based Method for Sumoylation Site Prediction

Christopher J. Friedline¹, Xueping Zhang², Zendra E. Zehner²,
and Zhongming Zhao^{1,3,*}

¹ Center for the Study of Biological Complexity, Virginia Commonwealth University,
Richmond, VA 23284, USA

² Department of Biochemistry, Virginia Commonwealth University, Richmond, VA 23298, USA

³ Departments of Psychiatry and Human Genetics, Virginia Commonwealth University,
Richmond, VA 23298, USA
zzhao@vcu.edu

Abstract. Post-translation modification by sumoylation is an important step in the regulation of many cellular processes. Existing programs predict sumoylation sites based on a consensus motif ϕ -K-X-E/D. However, ~23% of real SUMO sites do not match this motif, which makes prediction of sumoylation sites complicated. Here, we present a new method, FindSUMO, which predicts sumoylation sites using a position-specific scoring matrix. The comparison of FindSUMO with other two programs (SUMOsp and SUMOplot) indicates that FindSUMO accurately predicts sumoylation sites out of fewer total candidates and, in many cases, real sites with the highest scores. Overall, FindSUMO has at least the same performance as the other methods. While many improvements are expected, FindSUMO can also be applied to predict sumoylation sites that do not follow the consensus motif.

Keywords: SUMO; sumoylation; PSSM; position-specific scoring matrix.

1 Introduction

A variety of modifications occur at different levels in cell processes. Post-translational modification has been reported to be an indispensable biological process presented in cell signaling, death or localization. Sumoylation, a modification of lysine side chains in the target proteins by small ubiquitin-related modifier (SUMO), involves in many important cellular processes such as transcriptional regulation, transcription factor activity, and signal transduction [1-3].

SUMO proteins are highly conserved across eukaryotes. There are four SUMO family members in humans [4], three members in mice, two SUMO members in yeast, and at least eight SUMO paralogs in plants [5]. Many sumoylation modifications occur at a consensus amino acid motif ϕ -K-X-E/D (where ϕ represents the hydrophobic L, I, V or F and X represents any amino acid) [1,4]. However, previous work has shown that

* Corresponding author.

approximately 23% of real sumoylation sites do not match this consensus motif [6], which makes SUMO site prediction difficult.

There have been a few approaches to predicting sumoylation sites in proteins such as SUMOsp [6] and SUMOplot (<http://www.abgent.com/sumoplot>). These approaches usually predict sumoylation based on the consensus motif, thus they not only generate high false positive rates because many sites matching the consensus motif are not actually sumoylated, but also fail to identify the true positives that do not match the consensus motif. As a result, these approaches have limitations on accuracy and coverage in prediction.

To ease the limitation of prediction based on the ϕ -K-X-E/D motif, we applied a position-specific scoring matrix (PSSM) rather than pattern matching that is simply based on the specific sites within protein sequences. In a PSSM-based approach, an amino acid may be given different scores at different sites in protein sequences. The scores that are generated for a sub-region of a sequence can be either positive or negative, where a positive score represents a site at which the amino acid occurs more frequently than would be expected by chance and a negative score less frequently than would be expected by chance. Furthermore, our method (which we named FindSUMO) takes into account the information content present in each column of our PSSM, rather than simply using all columns present for a given length around a motif. This speeds up processing time since not all columns in the candidate motifs and matrix need to be searched, but rather only those that contain sufficient information.

2 Methods

2.1 PSSM-Based Method (FindSUMO)

We applied a PSSM to predict SUMO sites. Our matrix is built from a manually curated data set of known sumoylation sites. This list of known sites was based on the supplementary data set given by Xue et al. [6].

We retrieved the supplementary data and placed the information into a pre-formatted delimited text file, hereafter called the “mapping file.” Each line in the mapping file contained the protein accession number and coordinate location of the lysine residue present in the SUMO motif. For example, K84 indicates that the sumoylation site is identified by the lysine which is the 84th amino acid in the protein sequence. The sequences were downloaded from the NCBI protein database using Batch Entrez and stored in another file, hereafter called the “training file.” Records that were not found using the batch process, due to use of alternate IDs in the supplemental data set, were manually downloaded and added to the training file. The general algorithm is shown in Figure 1.

First, data from the mapping and training files are read by the computer program and motifs are extracted from the entire protein sequence using coordinates in the mapping file. The amino acids at each position in the motifs are counted, and the amount of information [7] present in each column was determined and stored if an arbitrary cutoff

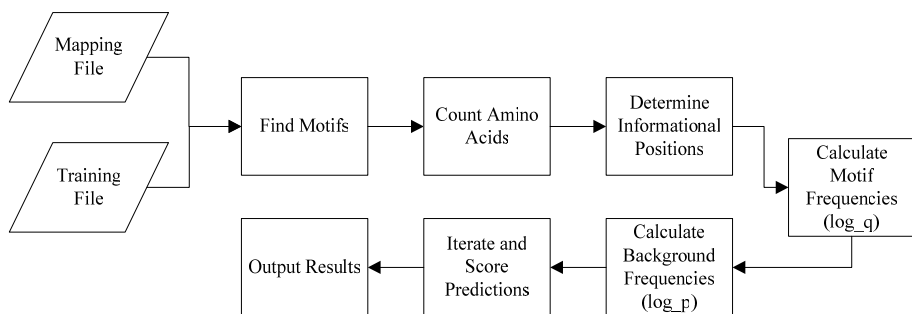


Fig. 1. Algorithm implemented to build a PSSM from mapping and training files

(0.6) is met. Next, for each informational column, the motif frequencies are determined as well as the background frequencies of each amino acid. With this now complete PSSM, the testing protein sequences are parsed and scored. If the scores meet or exceed a cutoff value, they are stored and output to files.

For each protein in the mapping file, the corresponding known SUMO site is extracted and stored using the mapping file coordinates and two parameters: the motif length, which we chose to fix at 7 to encompass the consensus motif plus some flanking region, and the lysine position in our mapping file. Once all of the motifs are extracted, the amino acids at each position are counted. To calculate the information contained in each column, we first determine the individual probabilities of the amino acids in each column. If an amino acid is not found in a column, its probability is set to zero. The column amino acid probabilities are then used to calculate the amount of uncertainty in a column (H_c) is given by equation 1:

$$H_c = -\sum_{i=1}^M P_i \log_2 P_i \quad (1)$$

Therefore, the information present in each column is given by the following equation describing information as the decrease in uncertainty [7]. Here, $M = 20$ because there are 20 possible amino acids. The formula for calculating the information content (R) is given in equation 2:

$$R = H_{before} - H_c = \log_2 M - H_c = \log_2 20 - H_c \quad (2)$$

If the information content in a given column is greater than our chosen cutoff value of 0.6, that column is stored for later use in the PSSM-based scoring of candidate sequences.

We then calculate the relative frequencies of amino acids for each informational column in the motif, using a pseudocount of 0.01 to avoid math errors and bias in situations where an amino acid is not found in a column, as shown in equation 3:

$$\log_{q,aa} = \ln(F_{aa}) = \ln \frac{(count_{aa} + (pseudocount * backgroundfrequency_{aa}))}{pseudocount + numberofsequences} \quad (3)$$

The background frequency of each amino acid is calculated according to equation 4 below using the sequences from our training data.

$$\log_{p,aa} = \ln(F_{bg,aa}) = \ln(backgroundfrequency_{aa}) \quad (4)$$

Now that these values are obtained for each amino acid in our training set, as calculated from the matrix, we score candidate protein sequences in a sliding window fashion of width equal to our motif length, starting at the first amino acid in our candidate sequence until the end. Each sub-region of the candidate is given a score based on the log-odds ratio of each amino acid in that informational position (\log_q) to the background frequency of that amino acid (\log_p). If that hit is greater than our cutoff value, we store that hit as a potential sumoylation candidate.

2.2 Method Evaluation

2.2.1 Motif Start Position

Our initial challenge was to determine the effect of the number and location of the informational columns in the PSSM on the ultimate prediction ability of the program. We did this by calibrating our program using the experimentally-verified data given by the $n = 242$ coordinates in the mapping file. To determine if our program could predict the verified sumoylation sites from our training data, the first step was to create separate FASTA-formatted files, each containing the protein sequences with the accession numbers as headers, for each training protein sequence. We set our program to use a score cutoff value of zero so that by including all positive hits we could maximize our chances of the program predicting a known site. We then chose increasing lysine positions from zero (not part of the motif) to nine (past the motif) in increments of one. For each lysine position, we kept track of the number of columns that contain information, the number of known SUMO sites from our training data that were predicted (“found”), the number of known SUMO sites that were not predicted (“not found”), and the percentage of known sites that were predicted.

2.2.2 Cutoff Value Maximization

With the start position determined, we processed each training sequence as before, but this time varying only the score cutoff, those values above which are included in our output, from 0 to 6 in 0.20 increments. First, we again discard all scores less than 0, allowing only scores which potentially can occur more frequently than expected by chance to be candidates for inclusion. Second, we calculate sensitivity (Sn), specificity (Sp), and accuracy (Ac), equations (5-7) for which are shown below.

$$Sn = \frac{TP}{TP + FN} \quad (5)$$

$$Sp = \frac{TN}{TN + FP} \quad (6)$$

$$Ac = \frac{TP + TN}{TP + FP + TN + FN} \quad (7)$$

We defined true positives as predictions above a cutoff value and also in our mapping data, false positives as predictions above a cutoff value and not in our mapping data, true negatives as predictions below a cutoff value and not in our mapping data, and false negatives predictions as predictions below a cutoff value and in our mapping data. We then plotted the probabilities associated with each performance metric against the tested cutoff values.

2.3 Testing Data

We obtained a group of 25 new protein sequences from the UniProt database [8]. These proteins have been experimentally verified for sumoylation sites recently, as confirmed by searching iHOP [9] for SUMO-1 and selecting target proteins uploaded between 2006 and 2008, and as such are not included in the training dataset. After this verification, we used our program to predict sumoylation target sites in 10 randomly chosen protein sequences, termed the “testing set,” comprising 12 known sumoylation sites from this group. For comparison, we also predicted sumoylation sites using two other programs: SUMOsp and SUMOplot.

3 Results and Discussion

3.1 Motif Start Position

Using the procedures in subsection 2.2.1 (e.g., increasing lysine position from zero to nine by an increment of one each time), we kept track of the number of columns that contain information and count the number of “found”. With start positions such that the lysine is either fourth or fifth, we were able to predict a known SUMO site 93% of the time (Table 1). Interestingly, although higher prediction percentages were found when the lysine is the sixth and seventh positions, the PSSM does not contain the maximum number of informational columns (i.e., the entire motif). Based on these results, we set the lysine at the fourth position to simultaneously maximize both prediction percentage and informational column number in the PSSM.

3.2 Cutoff Value Maximization

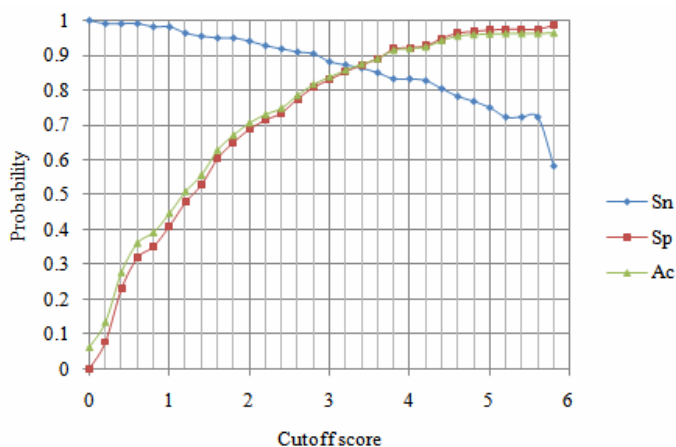
After setting the lysine at the fourth position, we calculated sensitivity, specificity, and accuracy using the training data. Figure 2 shows the distribution of these measurements using cutoff score values ranging from 0 to 6 in 0.20 increments (see subsection 2.2.2). We observed a trend that when cutoff value increases, sensitivity (and accuracy) increases but specificity decreases. The cutoff score value is 3.4, the point at which sensitivity and specificity first intersect. At this cutoff, $S_n = 0.864$, $S_p = 0.875$, and $A_c = 0.874$.

3.3 Performance of FindSUMO

We assessed FindSUMO using the testing dataset. Among the 10 proteins which contain 12 sumoylation sites, FindSUMO predicted 10 known sumoylation sites

Table 1. Results for varying lysine start positions

Position	Info. cols.	Found	Not found	Found (%)
0	1	190	52	78.5
1	2	227	15	93.8
2	3	222	20	91.7
3	3	223	19	92.2
4	3	225	17	93.0
5	3	225	17	93.0
6	2	232	10	95.9
7	2	231	11	95.5
8	1	182	60	75.2
9	0	0	242	0

**Fig. 2.** Performance evaluation with varying cutoff scores with length = 7 and start = 4

(10/12 = 83%). In the case where a prediction of a real site was made, it was predicted with the highest score 70% of the time.

3.4 Performance Comparison of FindSUMO with Other Two Programs

For the 12 sumoylation sites that we tested, there are several interesting features. One clear observation is that when our program could not predict a real SUMO site, all three programs failed (Table 2). Another equally important observation is that there is a protein for which our program is the only one able to make a correct prediction. There

are also cases that, although the real site was not predicted by any program, all three programs made another prediction. The Q8IZQ8 prediction exhibits this behavior.

We also found that when predictions were made by each program, they were given different rankings within each result set. For example, our program predicted the real K101 of Q04410 as the highest scoring possibility out of three potential sumoylation candidates for that protein, but SUMOsp predicted four possible SUMO sites and the real site was the least scoring hit. For this same protein, SUMOplot predicted seven total SUMO sites with the real site scoring the highest. We found similar scenario with Q9UQR1. Limited to these randomly selected test data, FindSUMO predicted the real site with the highest score 70% of the time, which is greater than both SUMOsp (38%) and SUMOplot (67%). Moreover, of those hits that were predicted less than the highest score, our program and SUMOplot found the same sites at the same rank.

The comparisons above indicate that FindSUMO, which is based on PSSMs, is a useful program for prediction of sub-regions of protein sequences that may be sumoylated. FindSUMO performs at least equally as well in locating known sumoylation sites as the other two methods described above. While our method also fails to predict the same real sumoylation sites as the other two programs, it predicts the real site with the highest score more frequently than the others.

With respect to the program itself, there are several advantages. First, we may extend PSSMs to any protein sequence motifs to identify functional regions for which there is sufficient experimental data. Second, the program, written in Perl, can be easily used

Table 2. Comparison of known SUMO sites that were predicted by three methods

UniProt#	Pos. of real K	Score (rank/total)		
		FindSUMO	SUMOsp 2.0	SUMOplot
Q60953	160	*	*	*
Q04110	5	7.2 (1/3)	4.972 (1/4)	0.94 (1/7)
Q04110	101	7.2 (1/3)	2.972 (4/4)	0.94 (1/7)
P12956	556	3.7 (3/6)	0.635 (4/6)	0.61 (3/12)
P17544	118	7.2 (1/2)	3.536 (1/2)	0.94 (1/4)
Q8IZQ8	445	*	*	*
Q9UQR1	115	6.8 (1/5)	3.464 (2/9)	0.93 (1/15)
Q9UQR1	356	6.8 (1/5)	2.848 (7/9)	0.93 (1/15)
Q07666	96	6.3 (1/3)	5.071 (1/3)	0.93 (1/7)
Q9UBW7	963	3.6 (3/5)	*	0.67 (3/13)
P04818	292	5.2 (2/3)	0.863 (2/2)	0.85 (2/6)
P00374	179	3.4 (1/1)	*	*

* Real K (lysine) not predicted by the program.

and modified in any platform. While all of the compared programs perform equally well, FindSUMO has the added benefits of supporting batch input of files while at the same time not needing a web server on which to run.

Although FindSUMO seems to be working well, many improvements are needed. For example, there is uncertainty whether the ranking of real hits within the total set of predictions is an indication of success. It might be possible that with different training data, the sites would be ranked differently in the output. We plan to test the effects of random sampling within the training set to construct the PSSM over many iterations to find any selection or ordering bias that effects our prediction rate or ranking. Another improvement is to incorporate structural information about these proteins to further rule out false positives, but the lack of such readily available information in computational form may pose a challenge.

References

1. Johnson, E.S.: Protein Modification by SUMO. *Annu. Rev. Biochem.* 73, 355–382 (2004)
2. Gill, G.: Post-translational Modification by the Small Ubiquitin-related Modifier SUMO Has Big Effects on Transcription Factor Activity. *Curr. Opin. Genet. Dev.* 13, 108–113 (2003)
3. Girdwood, D.W., Tatham, M.H., Hay, R.T.: SUMO and Transcriptional Regulation. *Semin Cell. Dev. Biol.* 15, 201–210 (2004)
4. Hay, R.T.: SUMO: A History of Modification. *Mol. Cell* 18, 1–12 (2005)
5. Kurepa, J., Walker, J.M., Smalle, J., Gosink, M.M., Davis, S.J., Durham, T.L., et al.: The Small Ubiquitin-like Modifier (SUMO) Protein Modification System in Arabidopsis. Accumulation of SUMO1 and -2 conjugates is increased by stress. *J. Biol. Chem.* 278, 6862–6872 (2003)
6. Xue, Y., Zhou, F., Fu, C., Xu, Y., Yao, X.: SUMOsp: A Web Server for Sumoylation Site Prediction. *Nucleic Acids Res.* 34, W254–W257 (2006)
7. Shannon, C.E.: The Mathematical Theory of Communication 1963. *MD Comput.* 14, 306–317 (1997)
8. Apweiler, R., Bairoch, A., Wu, C.H., Barker, W.C., Boeckmann, B., Ferro, S., et al.: UniProt: The Universal Protein Knowledgebase. *Nucleic Acids Res.* 32, D115–119 (2004)
9. Hoffmann, R., Valencia, A.: A Gene Network for Navigating the Literature. *Nat. Genet.* 6, 664 (2004)

Maximal-Robustness-Minimal-Fragility Controller: A Compromise between Robustness and Fragility of Biochemical Networks

Ming-Guang Shi^{1,2,3}, Michael R. Lyu⁴, and Tat-Ming Lok⁵

¹ Intelligent Computing Lab, Hefei Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei, 230031, China

smgsmg@mail.ustc.edu.cn

² Department of Automation, University of Science and Technology of China, Hefei, 230026, China

³ School of Electric Engineering and Automation, Hefei University of Technology of China, Hefei, 230009, China

⁴ Computer Science & Engineering Department, The Chinese University of Hong Kong

⁵ Information Engineering Department, The Chinese University of Hong Kong

Abstract. Establishing a trade-off between robustness and fragility has been an active research topic in constructing biochemical networks. In this paper, we formulate a compromise between robustness and fragility as a cooperative game, based on which a dynamical incomplete information game is constructed. In addition, three channels are chosen as players, and eight pure control strategies are created. Algorithms in seeking the perfect Bayesian-Nash equilibrium are consequently constructed. Based on the perfect Bayesian-Nash equilibrium, Maximal-Robustness-Minimal-Fragility controller (MRMFC) is derived, and MRMFC is effectively applied to biochemical networks. And computer simulations demonstrate that the biochemical network achieves a good balance between robust stability and dynamical performance. Consequently, an attractive solution in attacking the problem of the trade-offs between robustness and fragility is therefore laid out in this paper.

Keywords: $\mu/H_2/H_{\infty}$ control; Biochemical networks; Maximal-Robustness-Minimal-Fragility controller; Perfect Bayesian-Nash equilibrium.

1 Introduction

A system-level understanding of a biological system can be divided into two steps: (1) System modeling. It is meant to derive a mathematical model from the system structure. (2) Controller devising [1-2] It is meant to apply control theory to the biological mathematical model, and devise the negative controller to get better robustness and dynamical performance. In general, biological robustness is an essential property of biological systems [3-5]. The biological robustness is usually related to the feedback control [3-6]. The robustness of perfect adaptation is the result of the integral feedback control [5], and biological complexity is the interplay between complexity,

robustness, modularity, feedback and fragility [6]. In literature [7], a simple graphical method was presented to analyze the presence of multi-stability, bifurcations, and hysteretic behavior of positive-feedback systems.

The trade-offs among robustness, fragility and performance exist often in biological systems at different levels [8] while robustness and fragility are the important features of biological systems [9]. Robustness is generally viewed as robust stability while fragility is viewed as dynamic performance. While fragility is considered bad, the dynamic performance is deemed good. Therefore, it is a vital research problem that the biological system should achieve a good balance between robust stability and optimal dynamic performance. To solve this problem, we employ the game theory [10-11] to perform the task. Multi-objective control theory offers a very flexible design framework in which a control engineer can freely select arbitrary performance channels and uncertainty models, and the most appropriate norm to represent the design specification for each channel can be provided [12-14]. Mixed $\mu/H_2/H_\infty$ control incorporates all three control methods, i.e. μ control for improving the stability of uncertainty [15-17], H_2 control for improving dynamic performance, and H_∞ control for improving robust stability. We choose the structured singular value to represent the stability of uncertainty when we select μ control. Moreover, we employ H_2 norm to represent dynamic performance, H_∞ norm to represent the robust stability. Consequently, we can readily formulate a mixed $\mu/H_2/H_\infty$ control.

2 Mathematical Model

Since S-system [18] is a universal biological system, we transform it into linear system in order to construct robust control model. Generally, S-system is a type of power-law formalism. The equation of the S-system can incorporate the robustness and dynamical performance of a biological system. Using S-system steady-state evaluation, the control analysis of a given system can be easily established. Assuming reactant x_i is concentration, α_i and β_i are rate constants, g_{ij} and h_{ij} are the values of interactive effect of x_i and x_j , thus we can have [18,20]

$$\dot{x}_i = \alpha_i \prod_{j=1}^{n+m} x_j^{g_{ij}} - \beta_i \prod_{j=1}^{n+m} x_j^{h_{ij}} \quad (i = 1, 2, \dots, n) \tag{1}$$

Considering $\dot{x}_i = 0$, we can obtain:

$$\alpha_i \prod_{j=1}^{n+m} x_j^{g_{ij}} = \beta_i \prod_{j=1}^{n+m} x_j^{h_{ij}} \tag{2}$$

$$\ln \alpha_i + \sum_{j=1}^{n+m} g_{ij} \ln x_j - \ln \beta_i - \sum_{j=1}^{n+m} h_{ij} \ln x_j = 0 \tag{3}$$

$$\sum_{j=1}^n (g_{ij} - h_{ij}) \ln x_j + \sum_{j=n+1}^{n+m} (g_{ij} - h_{ij}) \ln x_j + \ln \left(\frac{\alpha_i}{\beta_i} \right) = 0 \tag{4}$$

Now, define $y_j = \ln x_j, a_{ij} = g_{ij} - h_{ij}, b_i = \ln(\frac{\alpha_i}{\beta_i})$ equation (4) can be denoted as

$AY + B_\omega \omega + B_u u = 0$, where

$$A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{nl} & \dots & a_{nn} \end{pmatrix} \quad B_u = \begin{pmatrix} a_{1n+k+1} & \dots & a_{1n+m} \\ \vdots & \ddots & \vdots \\ a_{nn+k+1} & \dots & a_{nn+m} \end{pmatrix} \quad B_\omega = \begin{pmatrix} a_{11} & \dots & a_{1n} & 1 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{nl} & \dots & a_{nn} & 0 & \dots & 1 \end{pmatrix} \quad (5)$$

We could obtain state-space equation when $X_g = Y : AX_g + B_\omega \omega + B_u u = 0$

And , the state-space equation could be derived when $\dot{x}_1 \neq 0$:

$$\begin{cases} F(\dot{x}_g) = Ax_g + B_\omega \omega_1 + B_u u \\ z_1 = C_1 x_g + D_{11} \omega_1 + D_{12} u \\ z_2 = C_2 x_g + D_{21} \omega_1 + D_{22} u \\ y = C_3 x_g + D_{31} \omega_1 + D_{32} u \end{cases} \quad (6)$$

$\omega_1 = [p \ \omega]^T, z_1 = q, z_2 = z, x_g$ is general state variable, $\Delta_m(s)$ is the worst case uncertainty $\Delta(s)$, ω is disturbance signal, u is control input signal, z is the evaluated output, y is the measured output, and q and p are input and output signals of $\Delta_m(s)$, respectively.

3 A Dynamical Incomplete Information Game

We formulate the trade-offs between robustness and fragility as a dynamical incomplete information game, and at the same time, choose channels T_{qp} , $T_{z\omega}$ and $T_{u\omega}$ as three players and eight control strategies as pure strategies, which are shown in Figs 1.

Three players could be constructed as follows: (1) Player1 refers to T_{qp} . From equation (6), it can be seen that T_{qp} is the transfer function from p , which is the output signal of $\Delta_m(s)$ to q with the input signal of $\Delta_m(s)$. (2) Player2 refers to $T_{z\omega}$. It can be seen that $T_{z\omega}$ is the transfer function from ω , which is disturbance signal to z with the evaluated output signal. (3) Player3 refers to $T_{u\omega}$. It can be readily seen that $T_{u\omega}$ is the transfer function from ω , which is disturbance signal to u with the control input signal.

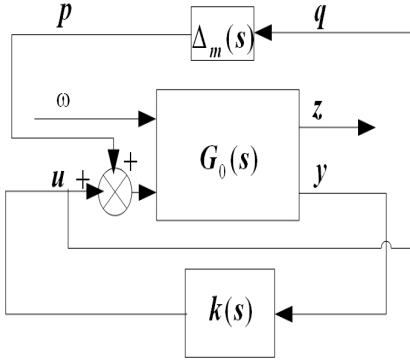


Fig. 1. Control of the generalized plant

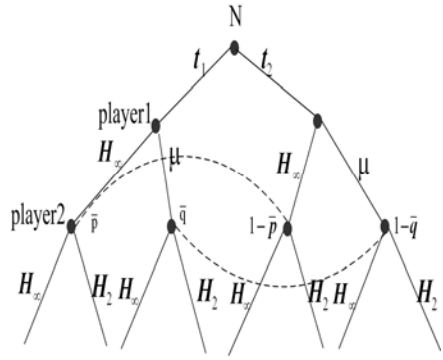


Fig. 2. The perfect Bayesian-Nash equilibrium

According to three controllers including μ 、 H_2 and H_∞ control, three players could select each of three controllers and design eight control strategies. Thus, eight control strategies and norms can be described as follows:

A1) H_∞ control .

A2) Mixed H_∞ / H_2 control. Minimizing $\|T_{u\omega}\|_2$ and $\left\| \begin{bmatrix} T_{qp} \\ T_{z\omega} \end{bmatrix} \right\|_\infty < \gamma$

A3) Mixed H_∞ / H_2 control. Minimizing $\|T_{z\omega}\|_2$ and $\left\| \begin{bmatrix} T_{qp} \\ T_{u\omega} \end{bmatrix} \right\|_\infty < \gamma$

A4) Mixed H_∞ / H_2 control. Minimizing $a\|T_{qp}\|_\infty^2 + b\left\| \begin{bmatrix} T_{qp} \\ T_{u\omega} \end{bmatrix} \right\|_2^2$

A5) Mixed μ / H_∞ control. Minimizing $\|T_{qp}\|_\mu$ and $\left\| \begin{bmatrix} T_{z\omega} \\ T_{u\omega} \end{bmatrix} \right\|_\infty$

A6) Mixed $\mu / H_\infty / H_2$ control. Minimizing $\|T_{qp}\|_\mu$ and $a\|T_{z\omega}\|_\infty^2 + b\|T_{u\omega}\|_2^2$

A7) Mixed $\mu / H_2 / H_\infty$ control. Minimizing $\|T_{qp}\|_\mu$ and $a\|T_{z\omega}\|_2^2 + b\|T_{u\omega}\|_\infty^2$

A8) Mixed μ / H_2 control. Minimizing $\|T_{qp}\|_\mu$ and $\left\| \begin{bmatrix} T_{z\omega} \\ T_{u\omega} \end{bmatrix} \right\|_2$

Generally, a payoff matrix is a table that describes the utility in a game for each possible combination of strategies. Therefore we obtain the payoff matrix of Tables I as follows. In the 3-person game, the row of a payoff matrix is player2's strategy and the

column of a payoff matrix is player3’s strategy when player1 chooses the strategy. For example, $\|\mathbf{T}_{z\omega}\|_\infty$ and $\|\mathbf{T}_{z\omega}\|_2$ are player2’s strategy, meanwhile, $\|\mathbf{T}_{u\omega}\|_\infty$ and $\|\mathbf{T}_{u\omega}\|_2$ are player3’s strategy from Table I. The values of matrix are the utility of player2、 player1 and player3, which are the reciprocal of their own norms.

Table 1. The payoff matrix when player1 chooses $f^1 = \{H_\infty \text{control}\} / f^2 = \{\mu \text{control}\}$

$\ \mathbf{T}_{z\omega}\ _\infty / \ \mathbf{T}_{z\omega}\ _\mu$	$\ \mathbf{T}_{u\omega}\ _\infty$	$\ \mathbf{T}_{u\omega}\ _2$
$\ \mathbf{T}_{z\omega}\ _\infty$	$(\frac{1}{b_{11}} \frac{1}{a_{11}} \frac{1}{c_{11}}) / (\frac{1}{b_{21}} \frac{1}{a_{21}} \frac{1}{c_{21}})$	$(\frac{1}{b_{12}} \frac{1}{a_{12}} \frac{1}{c_{12}}) / (\frac{1}{b_{22}} \frac{1}{a_{22}} \frac{1}{c_{22}})$
$\ \mathbf{T}_{z\omega}\ _2$	$(\frac{1}{b_{13}} \frac{1}{a_{13}} \frac{1}{c_{13}}) / (\frac{1}{b_{23}} \frac{1}{a_{23}} \frac{1}{c_{23}})$	$(\frac{1}{b_{14}} \frac{1}{a_{14}} \frac{1}{c_{14}}) / (\frac{1}{b_{24}} \frac{1}{a_{24}} \frac{1}{c_{24}})$

4 The Algorithms for Seeking Perfect Bayesian-Nash Equilibrium

We devise an algorithm to seek the perfect Bayesian-Nash equilibrium of a dynamical incomplete information game. In a cooperative dynamical incomplete information game, the utility depends on not only their own action, but the actions of others. Usually the perfect Bayesian-Nash equilibrium is the optimal strategy of players. No player can increase the system’s utility by unilaterally deviating from the perfect Bayesian-Nash equilibrium. When seeking the perfect Bayesian-Nash equilibrium, we begin with 2-person game (player1 and player2), which can be described in Fig.2. When this game is finished, we can get the Bayesian-Nash equilibrium of 2-person game. Considering player3’s strategy and 2-person Bayesian-Nash equilibrium we can obtain the Bayesian-Nash equilibrium of 3-person game.

Each player has its own strategy space $f^m = \{f_1^m, f_2^m\}$ which includes f_1^m and f_2^m . Therefore we can obtain strategy profiles, which include (f_1^m, f_1^m) 、 (f_1^m, f_2^m) 、 (f_2^m, f_1^m) and (f_2^m, f_2^m) ($m=1,2,3$). From Fig.2, it can be seen that \bar{p} and \bar{q} are player1’s posterior probabilities with $\bar{p} = \bar{p}(t_1 | H_\infty)$ and $\bar{q} = \bar{p}(t_1 | \mu)$ respectively. Meanwhile, we can get $1 - \bar{p} = \bar{p}(t_2 | H_\infty)$ and $1 - \bar{q} = \bar{p}(t_2 | \mu)$. We also define probabilities $p(t_1) = p(t_2) = \frac{1}{2}$ when we consider equilibrium state of t_1 and t_2 . $u_1(t_1, q_m, q_m)$ is player1’s utility function, where $q_m = \{\mu, H_\infty, H_2\}$ ($m=1,2,3$).

Algorithm 1. (Player 1 Selects $\{H_{\infty}\text{control}, H_{\infty}\text{control}\}$ or $\{\mu\text{control}, \mu\text{control}\}$)

Step 1: Posterior probabilities are defined as follows:

$$\bar{p}(t_1 | H_{\infty}) = \bar{p}(t_1 | \mu) = \frac{p(t_1)}{p(t_1) + p(t_2)}, \quad \bar{p}(t_2 | H_{\infty}) = \bar{p}(t_2 | \mu) = \frac{p(t_2)}{p(t_1) + p(t_2)}$$

Step 2: Judge the strategy of Player2.

Step2.1: Judge the strategy of Player2 when Player1 selects $\{H_{\infty}\text{control}, H_{\infty}\text{control}\}$.

$\sum_{t_k} \bar{p}(t_k | H_{\infty}) u_2(t_k, H_{\infty}, H_{\infty})$ is the player2's probabilistic utility function when Player2

selects the strategy of H_{∞} control. Meanwhile, $\sum_{t_k} \bar{p}(t_k | H_{\infty}) u_2(t_k, H_{\infty}, H_2)$ is

the player2's probabilistic utility function when Player2 selects the strategy of H_2 control.

Step2.2: Judge the strategy of Player2 when Player1 selects $\{\mu\text{control}, \mu\text{control}\}$

$\sum_{t_k} \bar{p}(t_k | \mu) u_2(t_k, \mu, H_{\infty})$ is the player2's probabilistic utility function when Player2

selects the strategy of H_{∞} control. Meanwhile, $\sum_{t_k} \bar{p}(t_k | \mu) u_2(t_k, \mu, H_2)$ is

the player2's probabilistic utility function when Player2 selects the strategy of H_2 control.

Rule1.1: If $\sum_{t_k} \bar{p}(t_k | H_{\infty}) u_2(t_k, H_{\infty}, H_{\infty}) > \sum_{t_k} \bar{p}(t_k | H_{\infty}) u_2(t_k, H_{\infty}, H_2)$, then Player2

selects H_{∞} control; otherwise, Player 2 selects H_2 control.

Rule1.2: If $\sum_{t_k} \bar{p}(t_k | \mu) u_2(t_k, \mu, H_{\infty}) > \sum_{t_k} \bar{p}(t_k | \mu) u_2(t_k, \mu, H_2)$, then Player2

selects H_{∞} control; otherwise, Player2 selects H_2 control.

Step 3: Judge the strategy of Player1 on account of Player2's strategy.

Rule1.3: If $u_1(t_1, H_{\infty}, H_{\infty}) > u_1(t_1, H_{\infty}, H_2)$ and $u_1(t_2, H_{\infty}, H_{\infty}) > u_1(t_2, H_{\infty}, H_2)$, then Player1 selects H_{∞} control and Player2 selects H_{∞} control.

Rule1.4: If $u_1(t_1, H_{\infty}, H_2) > u_1(t_1, H_{\infty}, H_{\infty})$ and $u_1(t_2, H_{\infty}, H_2) > u_1(t_2, H_{\infty}, H_{\infty})$, then Player1 selects H_{∞} control, and Player2 selects H_2 control.

Rule1.5: If $u_1(t_1, \mu, H_{\infty}) > u_1(t_1, \mu, H_2)$ and $u_1(t_2, \mu, H_{\infty}) > u_1(t_2, \mu, H_2)$, then Player1 selects μ control and Player 2 selects H_{∞} control.

Rule1.6: If $u_1(t_1, \mu, H_2) > u_1(t_1, \mu, H_\infty)$ and $u_1(t_2, \mu, H_2) > u_1(t_2, \mu, H_\infty)$, then Player1 selects μ control and Player2 selects H_2 control.

B. Algorithm 2 (Player1 Selects the Strategy of $\{\mu$ control, H_∞ control $\}$). The implementation is just like *Algorithm 1*

C. Algorithm 3 (Player1 Selects the Strategy of $\{H_\infty$ control, μ control $\}$)

Step1: If H_∞ control is the optimal strategy of t_1 , then H_∞ control is not the optimal strategy of t_2 . If μ control is the optimal strategy of t_2 , then μ control is not the optimal strategy of t_1 . we can define $\bar{p}(t_1|H_\infty)=\bar{p}(t_2|\mu)=1$ and $\bar{p}(t_2|H_\infty)=\bar{p}(t_1|\mu)=0$
Steps 2 and 3 is like Steps 2 and 3 of Algorithm 1.

5 Experimental Results

Considering the glycolytic-glycogenolytic pathway in rat liver, the kinetic properties of the pathway can be described as shown in the following equations [18]:

$$\begin{aligned} \dot{x}_1 &= 0.077884314x_4^{0.66} - 1.062708258x_1^{1.53}x_2^{-0.59}x_7 \cdot \\ \dot{x}_2 &= 0.585012402x_1^{0.95}x_2^{-0.41}x_5^{0.32}x_7^{0.62}x_{10}^{0.38} - 7.93456 \times 10^{-4}x_2^{3.97}x_3^{-3.06}x_8 \quad (7) \\ \dot{x}_3 &= 7.93456 \times 10^{-4}x_2^{3.97}x_3^{-3.06}x_8 - 1.05880847x_3^{0.3}x_9 \end{aligned}$$

Diseases always alter the kinetic properties of a biochemical network. Suppose the pathway suffers from a parameter perturbation, which can be described as follows:

$$\begin{aligned} \dot{x}_1 &= 0.077884314x_4^{0.66} - 1.062708258x_1^{1.5135}x_2^{-0.6278}x_7 \cdot \\ \dot{x}_2 &= 0.585012402x_1^{0.9133}x_2^{-0.437}x_5^{0.32}x_7^{0.62}x_{10}^{0.38} - 7.93456 \times 10^{-4}x_2^{3.9125}x_3^{-3.035}x_8 \quad (8) \\ \dot{x}_3 &= 7.93456 \times 10^{-4}x_2^{3.9125}x_3^{-3.035}x_8 - 1.05880847x_3^{0.3825}x_9 \end{aligned}$$

$$x_6 = 5.02, x_9 = 4909, x_{10} = 2.04 * 10^{-9} \quad (\text{i.e. } A=A_0 + \Delta A, \Delta_m(s) = \Delta A)$$

Denoting $\gamma = 1$, $a=0.5$, $b=0.5$, and results is as follows: Player1 selects μ control, Player2 selects H_2 control and Player3 selects H_∞ control (2) Player1 selects H_∞ control, Player2 selects H_∞ control and Player3 selects H_∞ control.

It is shown that this system is stable and can converge to the steady state when there exists the uncertainty disturbance $\Delta_m(s)$ and noise disturbance ω from Fig. 3 to Fig. 5. The $\mu/H_2/H_\infty$ controller produces good robustness of the biochemical network

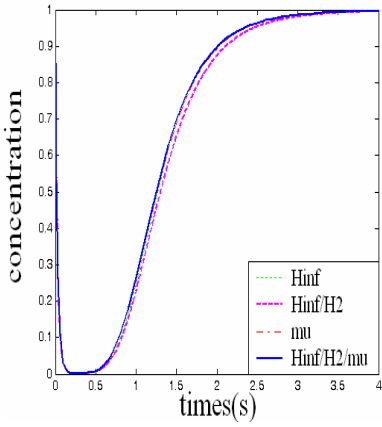


Fig. 3. The concentration of glucose-1-P

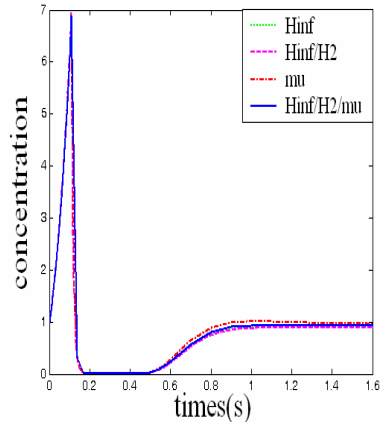


Fig. 4. The concentration of glucose-6-P

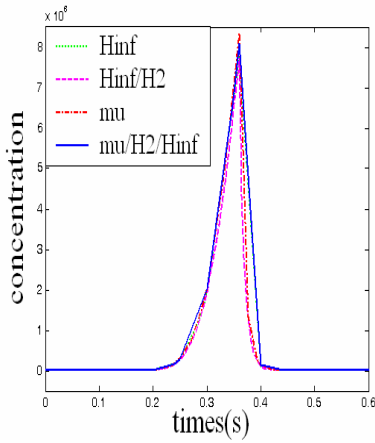


Fig. 5. The concentration of fructose-6-P

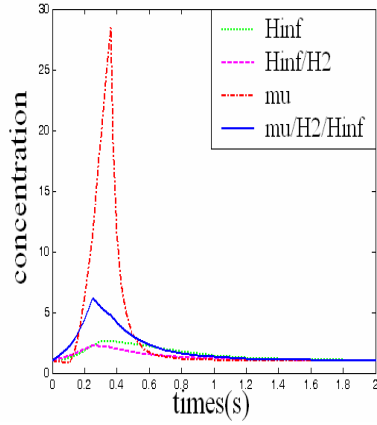


Fig. 6. The concentration of P_i

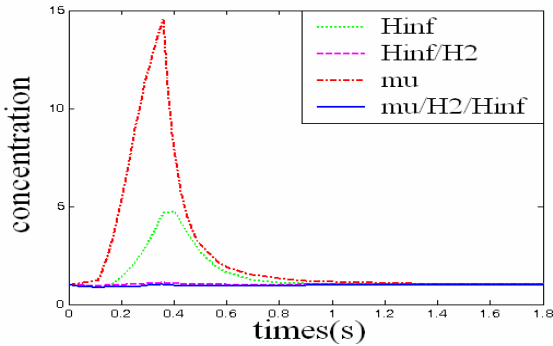


Fig. 7. The concentration of glucose

and robustness provides insensitivity to the parameter changes in this biochemical network. The poles of $\mu/H_2/H_\infty$ controller $K_{\mu/H_2/H_\infty}$ are -17.2360 , $-0.2518+3.0200i$, and $-0.2518-3.0200i$, respectively, so the controller $K_{\mu/H_2/H_\infty}$ is a stable controller. The concentrations of glucose-1-P, glucose-6-P and fructose-6-P with $\mu/H_2/H_\infty$ controller achieve a better dynamical performance and the peak values of the concentration of glucose-1-P, glucose-6-P and fructose-6-P are lower than those of other controllers as shown from Fig 3 to Fig 5. Convergent rate is rapid and convergent time is short when we use $\mu/H_2/H_\infty$ controller. It is shown that mixed H_∞/H_2 control has the worst performance from Fig 3 to Fig 4 and μ control has the worst performance from Fig 5. From Fig 6, the biochemical network achieves better disturbance attenuation performance specification with mixed H_∞/H_2 control controller when the input value is identical. Also from Fig 7, it can be seen that the biochemical network achieves better disturbance attenuation performance specification with $\mu/H_2/H_\infty$ controller when the input value is identical. In conclusion, $\mu/H_2/H_\infty$ controller, which is Maximal-Robustness-Minimal-Fragility controller, outperforms other controllers and the biochemical network achieves a good balance between robust stability and dynamical performance. Maximal-Robustness-Minimal-Fragility negative feedback controller is helpful for maintaining stability margin and disturbance attenuation when uncertainty disturbance and noise disturbance are considered. This achievement is important to biochemical network, as errors in signal transduction can result in growth impairment or cancer [19].

6 Conclusions

Generally, S-system is a popular biochemical networks model, so we transform it into robust control model. The trade-offs between robustness and fragility is formulated as a dynamical incomplete information game model. We seek the perfect Bayesian-Nash equilibrium and devise the Maximal-Robustness-Minimal-Fragility controller when this control model has noise disturbance and uncertainty disturbance. Biochemical network needs to work in robust state against the various sources of inter- or intracellular perturbations. The biological robustness and dynamical performance provide general principle for understanding many biological phenomena and for constructing a system-level view of medical therapy and disease.

Acknowledgments. This work was supported by the grant of Sino-Korea international cooperation project, No.30711140358, the grant of the Guide Project of Innovative Base of Chinese Academy of Sciences (CAS), No.KSCX1-YW-R-30, the grant of Oversea Outstanding Scholars Fund of CAS, No.2005-1-18.and HFUT, No.070403F.

References

1. Kitano, H.: Systems Biology: A Brief Overview. *Science* 295, 1662–1664 (2002)
2. Kitano, H.: Computational Systems Biology. *Nature* 420, 206–210 (2002)
3. Barkai, N.: Robust in Simple Biochemical Networks. *Nature* 387, 913–917 (1997)

4. Alon, U., Surette, M.G., Barkai, N., Leibler, S.: Robustness in Bacterial Chemotaxis. *Nature* 397, 168–171 (1999)
5. Yi, T.M., Huang, Y., Simon, M.I., Doyle, J.: Robust Perfect Adaptation in Bacterial Chemotaxis through Integral Feedback Control. *PNAS* 97, 4649–4653 (2000)
6. Csete, M.E., Doyle, J.: Reverse Engineering of Biological Complexity. *Science* 295, 1664–1669 (2000)
7. Angeli, D.: Detection of Multistability, Bifurcations, and Hysteresis in a Large Class of Biological Positive-Feedback Systems. *PNAS* 101, 1822–1827 (2004)
8. Kitano, H.: Biological Robustness. *Nature* 5, 826–837 (2004)
9. Carlson, J.M., Doyle, J.: Complexity and Robustness. *PNAS* 99, 2538–2545 (2002)
10. Basar, T., Olsder, G.J.: *Dynamic NonCooperative Game Theory*. Academic Press, New York (1982)
11. Fudenberg, D., Tirole, J.: *Game Theory*. MIT Press, Cambridge (1992)
12. Isidori, A.: *Mixed H₂ / H_∞ Control*. Springer, Heidelberg (1995)
13. Bernstein, D.S., Haddad, W.M.: LQG Control with a H_∞ Performance Bound: A Riccati Equation Approach. *IEEE Transactions on Automatic Control* 34, 293–305 (1989)
14. Zhou, K., Glover, K., Bodenheimer, B., Doyle, J.: Mixed H₂ and H_∞ Performance Objectives I: Robust Performance Analysis. *IEEE Transactions on Automatic Control* 39, 1564–1574 (1994)
15. Packard, A., Doyle, J.: The Complex Structured Singular Value. *Automatica* 29, 71–109 (1993)
16. Lin, J.L., Postlethwaite, I., Gu, D.W.: μ -K-Iteration for μ -Synthesis. *Automatica* 29, 219–224 (1993)
17. Balas, G.J., Doyle, J., Glover, K., Packard, A., Smith, R.: μ -Analysis and Synthesis Toolbox. *Automatica* 30, 733–735 (1994)
18. Voit, E.O.: *Computational Analysis of Biochemical System*. Cambridge University Press, Cambridge (2000)
19. Markus, K., Linda, L., Kilian, B., Jens, T., Victor, S.: Design Principles of a Bacterial Signaling Network. *Nature* 438, 504–507 (2005)
20. Chen, B.S., Wan, Y.C., Wu, W.S., Li, W.H.: A New Measure of the Robustness of Biochemical Networks. *Bioinformatics* 21, 2698–2705 (2005)

Predict Molecular Regulatory Network of Norway Rat under the Frame of Data Integration

Qiguo Rong and Zhe Zhang

Biomedical Engineering, College of Engineering, Peking University,
100871 Beijing, People's Republic of China
qrong@pku.edu.cn, zhangzhe.pku@gmail.com

Abstract. Build rat's molecular regulatory network by integrating five heterogeneous data types that serve as evidence for either protein-protein interaction or protein-DNA interaction. P-values for evidence types are calculated by different algorithms and merged together by Support Vector Machines according to estimated weights which indicate respective contributions of different evidence types to the final prediction. Proper classification threshold is specified to effectively control the false discovery rate, and the result is validated by searching predicted interactions in related databases as well as projecting them to signaling pathways to mark up key factors in disease mechanism. An analysis of our methodology versus previous studies and data integration versus single evidence is performed to demonstrate that the solution we present here is more comprehensive and advantageous than traditional ones due to its rational frame structure and full use of information.

Keywords: data integration; evidence; molecular regulatory network; protein-protein interaction; protein-DNA interaction; support vector machines.

1 Introduction

To unfold the whole gene map in front of human beings and shed light on the cure of human diseases, systems biology has been built up by modern biologists to deeply investigate the molecular regulatory network (MRN) of creatures by exploring dynamic characteristics at genomic/proteomic level [1].

Three types of data are widely used in molecular biology research: high-throughput experiment data such as microarray, existing knowledge proven by previous experiments or researches, predictions result from newly developed algorithms [2,3]. Each type of data has its own intrinsic drawbacks, as a result, any model basing on single data source may probably be partial or even misleading. Therefore, there is an urgent call for us to substitute traditional methods with a more integrative and comprehensive one. Systems biology presents itself under such circumstances. It integrates various types of data (also called evidence) by employing proper modeling tools such as Bayesian Network [4], Probabilistic Decision Tree, Support Vector Machines (SVM) [5], etc. This strategy makes full use of information from diverse aspects and allows different evidence types to make up shortcomings for each other.

In April 2004, NATURE announced the genome sequencing result of Norway Rat. The fact that 90% of rat's genes have matches in human and mouse makes rat one of the most important model organisms to study human diseases [6]. This article tries to figure out a systematic way to construct MRN of rat with detailed methodology given. Under data integration frame, five types of evidence are prepared to provide useful information from different viewpoints and SVM is introduced to merge them according to estimated weights. Compared with traditional methods, a more comprehensive view of gene map is promised by the diversity of data sources as well as SVM's supervised learning capability and good control of false discovery rate.

2 Methods

Protein-protein interaction (PPI) and protein-DNA interaction (PDI) are two of the most important molecular interactions in cells. PPI occurs almost everywhere in cells transferring biological signals, while PDI mainly occurs in nucleus as a transcriptional level interaction between transcription factors (TFs) and DNA molecules. The predicted MRN would be more convincing if more than one evidence types are found to indicate the high possibilities of PPI/PDI's existence among the molecules contained in the network. Five types of evidence are selected either from PPI or PDI perspective in this article: (1) Pearson Correlation Coefficient (PCC) calculated from microarray as evidence for PDI, (2,3) supervised classification results inferred by SVM as evidence for PDI and PPI, (4) subcellular localization as evidence for PPI, (5) homologous alignment among rat, human and mouse as evidence for PPI and PDI.

2.1 Data Preprocessing

324 Affymetrix RG-U34A gene array samples are downloaded from GEO database of NCBI. Then, microarray normalization is carried out to eliminate the noise to improve data quality. Firstly, Total Intensity Normalization is performed to make different samples comparable, secondly, Locally Weighted Linear Regression is conducted to clean up systematic bias which is the major source of deviation. Both normalization methods are implemented by an R package of Bioconductor.

There are 5152 genes with symbols On Affymetrix RG-U34A gene array. However, only those differentially expressed genes who have bigger chance to be involved in PPIs or PDIs will be taken into account, because this can not only avoid unnecessary distraction but also lighten computation burden in order to increase efficiency of calculation [7]. Among the 5152 genes having symbols, 2513 are assigned by the software of Significance Analysis of Microarrays (SAM) as significantly differentially expressed with the false discovery rate of 3.225%. The integration frame is established just on these 2513 genes and there are totally 3158841 interaction candidates to be investigated.

2.2 Pearson Correlation Coefficient

PCC is a popular measurement of how much two molecules relate with each other [8]. If one gene regulates another, the absolute value of PCC measuring the similarity

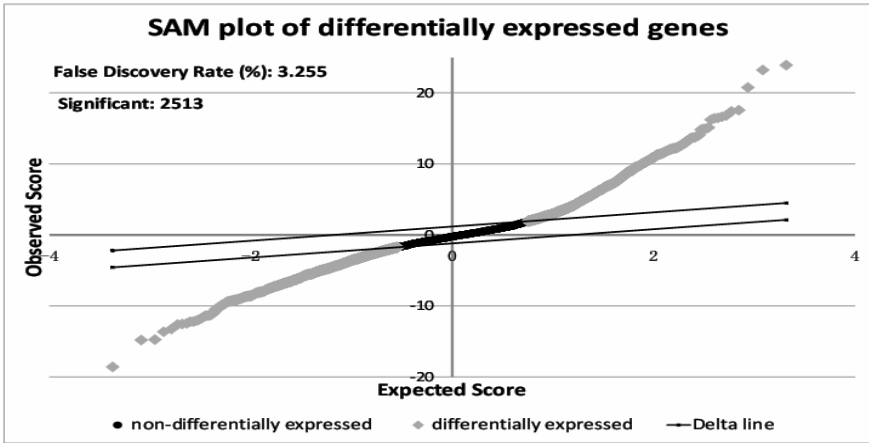


Fig. 1. Differentially expressed genes are assigned by SAM. The dark dots lying between delta lines represent regularly expressed genes, and the other 2513 dots marked in light gray are differentially expressed. The false discovery rate is merely 3.225% which means that almost all differentially expressed genes are selected out with a very small chance of mistaking.

between their expression patterns should be closer to 1 than the average. For all interaction candidates, PCCs are calculated and transformed to probability densities by Gaussian Kernel Density Transformation (GKDT) which scales all PCCs in a more statistical way to facilitates the calculation of P-values. Gaussian kernel density of *i*th interaction candidate *gkd_i* is computed as follows:

$$gkd_i = f(pcc_i) = \frac{\sum_{pcc_k \in \Delta(pcc_i)} \frac{1}{\sqrt{2\pi}\lambda} \exp\left\{-\frac{(pcc_k - pcc_i)^2}{2\lambda^2}\right\} * pcc_k}{N * \lambda} \quad (1)$$

In the formula above, $\Delta(pcc_i)$ is a neighborhood area with a fixed width of λ around pcc_i , and N indicates the number of pcc_k located in this neighborhood area. P-values are calculated according to Gaussian kernel densities.

2.3 SVM Classification

Being a supervised learning algorithm, SVM takes advantage of prior knowledge of confirmed PPIs/PDIs to classify the unknown ones [5]. The main idea is to transform Nonlinear separable issues to linear separable ones in high-dimensional space, where an optimal interface between interactions and non-interactions is found. This character differentiates SVM from the linear solution of PPC.

For PPI prediction, 2068 known PPIs with high credibility are downloaded from BIND. Part of them are assigned as positive samples in the training dataset, and the others are left to do the validation. Besides, a series of parameters have to be specified to ensure that the SVM is well trained by but not over fits to the training datasets. The input of the SVM classifier is a vector x composed of expression values of two genes

and a binary which is either 1 or -1 indicating whether there is interaction between them. A weight vector w and a bias constant b are estimated after training, and the predicted value for the i th candidate is $svm_i = w \cdot x_i + b$. All of the predicted values are finally transformed to P-values by GKDT mentioned above. The recall of our SVM is 90.6% which is high enough to guarantee that most true positive samples are correctly predicted, and the precision is 67.7% which is not too strict to prevent discovering new interactions from the negative set. The process of PDI prediction is basically the same as PPI with only two major differences: known PDIs are downloaded from TRED which stores almost all interactions between TFs and genes so far; only the proteins located in nucleus are under consideration because or else there is little chance for them to get in touch with the genes.

2.4 Subcellular Localization

Gene ontology is an indispensable tool to do clustering, classification, MRN inferring, etc. Cellular component as an important subcategory of GO is integrated to our frame because a lot of investigation shows that the possibility of interaction increases as two proteins get closer to each other [9]. Distance between proteins can be measured by Minimum Number of Transport Process (MNTP) which calculates the number of membranes that one protein has to pass through to meet the other. The smaller MNTP is, and the more likely two proteins are going to have physical interaction.

The selected 2513 genes are submitted to Princeton University’s GO query system to be grouped into 26 different subcellular localizations including Cytoplasm, Golgi apparatus, Nucleus, etc. The MNTP for each pair of subcellular localizations is estimated and then the P-value for $MNTP = k$ ($k = 0, 1, 2 \dots$) is computed as follows:

$$P_{MNTP=k} = 1 - \sum_{i \geq k} \frac{N_{MNTP=i} (N_{MNTP=i} - 1) / 2}{N_a (N_a - 1) / 2} \approx 1 - \sum_{i \geq k} \frac{N_{MNTP=i}}{N_a} \tag{2}$$

$N_{MNTP=k}$ denotes the number of protein pairs whose MNTPs are k . In this article $N_{MNTP=k}$ is approximated by the number of proteins that can be found in k different cellular components for convenience. N_a is the total number of proteins. When MNTP equals 0 which means protein pairs locate in the same cellular component, P-value has to be manually assigned to be 0.05. If we are not sure about the distance between two subcellular localizations, P-value is assigned to be 0.95 for conservative consideration. P-values for all possible MNTPs are shown in Table 1 in detail.

Table 1. P-values for MNTPs which measure distance between proteins

MNTP	N_{MNTP} (estimated)	P-value
0	696	0.05
1	312	0.4974
2	370	0.5834
3	491	0.6855
4	402	0.8209
5	167	0.9318
Not sure	-	0.95

2.5 Homologous Alignment

The assumption behind Homologous alignment is straightforward that the closer two mammals are on the evolutionary tree, the more genomic/proteomic similarities they have [10]. As we can see in the following figure, Human, mouse and rat (underlined) come from the same major clade. Intuitively, PPIs and PDIs can be inferred by comparing rat’s interactome with that of human and mouse. A name matching is performed between the 3158841 interaction candidates and all known interactions of human and mouse downloaded from BIND and TRED. As long as some candidate has interlog in human or mouse, its P-value is assigned to be 0.05 which is usually taken as the significance threshold to refute the null hypothesis.

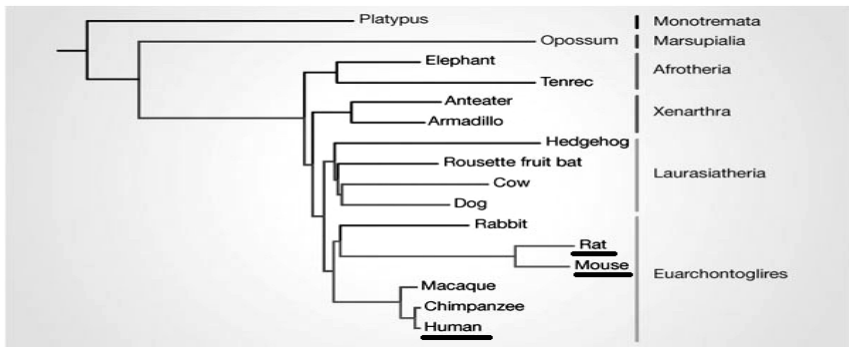


Fig. 2. Evolutionary tree of mammals cited from Nature [10]. Rat, mouse and human come from the same clade and are underlined to mark their Physiological closeness.

3 Results

In order to integrate multiple evidence types in a more objective manner, SVM is carried out again learning from foregone knowledge to estimate coefficients for five evidence types. Table 2 presents the final weights transformed from SVM coefficients to show how much each type of evidence contributes to the final prediction. Obviously, SVM for PPI and homologous alignment are the most dominant because the sum of their weights reaches 0.76. As for subcellular localization, PCC and SVM for PDI, the low level of importance may be due to inherent lack of valuable information or overlapping with other types of evidence.

Table 2. Weights of evidence types estimated by SVM

Evidence	SVM Coefficient	Weight of Evidence
PCC	-0.056	0.138
SVM for PPI	-0.101	0.449
SVM for PDI	0.011	0.070
Subcellular localization	-0.027	0.032
Homologous alignment	-0.084	0.311

Evidence types are integrated according to their weights for all candidates. Under conservative consideration, the classification threshold is set as 0.001 which is comparatively small to avoid large number of false positives. 3282 candidates are predicted true accordingly, and 31 of them overlap with records in BIND and TRED. To further validate our predicted MRN, APID2NET, a plug-in of Cytoscape which collects all PPIs stored in BIND, BioGrid, DIP, HPRD, IncAct and MINT is introduced, and another 19 candidates are verified to be true positives as a result. The network composed of these 50 true positives is shown in Fig. 3 with its layout automatically given by Cytoscape. Consequently, The true positive rate (TPR) also known as sensitivity is 9.1%, and the false positive rate (FPR) is 0.1% which equals 1-specificity. TPR can be further increased by setting higher threshold; however, FPR will rise as a sacrifice. Researchers should determine the threshold according to the dynamic balance between TPR and FPR as well as their specific requirements.

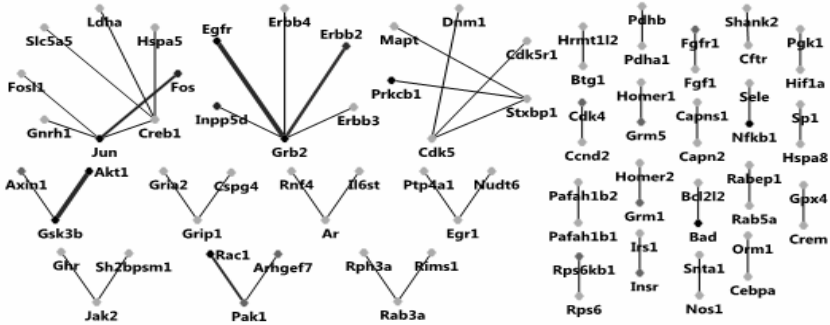


Fig. 3. Predicted MRN validated by authorized databases at the significance level of 0.001. Each node corresponds to a specific gene and each line indicates that there is interaction between the two nodes it connects. The 50 validated interactions are projected to pathways stored in KEGG by Pathway-Express. Node darkness and Line width are proportional to the appearance frequency of corresponding genes and interactions in all related pathways.

One of the goals of systems biology is to find out the relationship between molecular interactions and macroscopic phenotypes. Therefore, the final step of our method is to project predicted interactions to signaling pathways stored in KEGG to mark key factors in the mechanism of diseases. Pathway-Express is a component of the Onto-Tools ensemble and is designed to provide a system automatically finding well-known pathways involving our predicted interactions. Among the 53 pathways found to be associated with the 50 validated interactions, ErbB signaling pathway, MAPK signaling pathway and prostate cancer have the smallest P-values which implies that they have the closest relations with some of our predicted interactions. In Fig. 3, node darkness and line width are proportional to the frequency of finding corresponding gene and interaction in all signaling pathways respectively.

4 Discussion

The five evidence types are organized in parallel under our data integration frame. By comparison, a previous research which similarly combined evidence from six datasets

divided all protein pairs into 2^6 subsets according to combinations of evidence types and estimated error rates for each subset [11]. However, it performs poorly because the burden of calculating error rates grows exponentially as the number of evidence types increases. Besides, huge number of parameters induces potential over-fitting. These problems are tackled by our research in which all evidence types are processed separately and then merged together by specific weights inferred from their relative contributions to the final prediction. It is flexible for us to add or remove evidence with limited change of parameter number when the evidence weight is too big or too small. To allow more evidence types incorporated, Jansen et al. published a related study using Naïve Bayes method assuming conditional independency between evidence types [4]. However, this is not usually the case even if they proved the lack of linear correlation between most of the evidence types. Our study handles this problem by estimating weights for different types of evidence with SVM no matter whether there is dependency between them or not. SVM captures linear or nonlinear correlation between evidence types, as a result, the loss of one weight is compensated by another's increase to keep their total influence unchanged.

Table 3. Compare predicting performance of data integration with that of single evidence

Evidence	TPR	FPR
5 evidences integration	9.1%	0.102%
Pearson Correlation	3.1%	0.104%
SVM for PPI	5.5%	0.103%
Homologous Comparison	4.5%	0.103%

Does data integration really outperform single evidence? Table 3 indicates that the major difference between them is TPR. Data integration has the highest TPR of 9.1% which means it is able to find out the most true molecular interactions among the four listed methods. This character makes it more reliable than single evidence to explore new regulatory relationships in the unknown world of molecular biology. TPR of SVM for PPI is substantially higher than the other two types of single evidence which might be one of the reasons that it has the highest weight out of five. The following conclusions can be drawn from above analysis that data integration indeed overmatches single evidence regarding our research due to the fact that it allows different evidence types to complement each other to lead to a more integrative and comprehensive solution.

5 Conclusions

Data integration is becoming prevalent in the field of systems biology. In our research, five evidence types are selected to construct rat's MRN from perspectives of PPIs and PDIs. On one hand, different types of evidence make up shortcomings for each other; on the other hand, SVM is employed to make use of existing knowledge to the largest extent. Both of the above jointly promise a high predicting sensitivity as well as a good control of FPR. The approach we present here provides testable hypotheses for high-throughput experimental validation and a systematic way to

discover more molecular interactions efficiently. This general approach can be readily applied to other characteristics of gene or protein pairs and in other organisms as large-scale genomic/proteomic information becomes available.

References

1. Kitano, H., et al.: Systems Biology: A Brief Overview. *J. Science* 295, 1662–1664 (2002)
2. Hwang, D., Smith, J.J., Leslie, D.M., Weston, A.D., Rust, A.G., Ramsey, S., de Atauri, P., Siegel, A.F., Bolouri, H., Aitchison, J.D., Hood, L.: A Data Integration Methodology for Systems Biology. *J. PNAS* 102, 17296–17301 (2005)
3. Hwang, D., Smith, J.J., Leslie, D.M., Weston, A.D., Rust, A.G., Ramsey, S., de Atauri, P., Siegel, A.F., Bolouri, H., Aitchison, J.D., Hood, L.: A Data Integration Methodology for Systems Biology: Experimental Verification. *J. PNAS* 102, 17302–17307 (2005)
4. Jansen, R., Yu, H., Greenbaum, D., Kluger, Y., Krogan, N.J., Chung, S., Emili, A., Snyder, M., Greenblatt, J.F., Gerstein, M.: A Bayesian Networks Approach for Predicting Protein-Protein Interactions from Genomic Data. *J. Science* 302, 449–453 (2003)
5. Pavlidis, P., Weston, J., Cai, J., Noble, W.S.: Learning Gene Functional Classifications from Multiple Data Types. *J. Comput Biol.* 9, 401–411 (2002)
6. Rat Genome Sequencing Project Consortium.: Genome Sequence of the Brown Norway Rat Yields Insights into Mammalian Evolution. *J. Nature* 428, 493–521 (2004)
7. Pan, W.: A Comparative Review of Statistical Methods for Discovering Differentially Expressed Genes in Replicated Microarray Experiments. *J. Bioinformatics* 18, 546–554 (2002)
8. Lu, L.J., Xia, Y., Paccanaro, A., Yu, H.Y., Gerstein, M.: Accessing the Limits of Genomic Data Integration for Predicting Protein Networks. *J. Genome Res.* 15, 945–953 (2005)
9. von Mering, C., Krause, R., Snel, B., Cornell, M., Oliver, S.G., Fields, S., Bork, P.: Comparative Assessment of Large-Scale Data Sets of Protein-Protein Interactions. *J. Nature* 417, 399–403 (2002)
10. Kerstin, L.T.: Genome Sequencing: Three's Company. *J. Nature* 428, 475–476 (2004)
11. Jansen, R., Lan, N., Qian, J., Gerstein, M.: Integration of Genomic Datasets to Predict Protein Complexes in Yeast. *J. Structural and Functional Genomics* 2, 71–81 (2002)

Using Cost-Sensitive Learning to Determine Gene Conversions

Mark J. Lawson, Lenwood Heath, Naren Ramakrishnan, and Liqing Zhang

Department of Computer Science, Virginia Tech., USA
{malawso4,heath,naren,lqzhang}@vt.edu

Abstract. Gene conversion, a non-reciprocal transfer of genetic information from one sequence to another, is a biological process whose importance in affecting both short-term and long-term evolution cannot be overemphasized. Knowing where gene conversion has occurred gives us important insights into gene duplication and evolution in general. In this paper we present an ensemble-based learning method for predicting gene conversions using two different models of reticulate evolution. Since detecting gene conversion is a rare-class problem, we implement cost-sensitive learning in the form of a generated cost matrix that is used to modify various underlying classifiers. Results show that our method combines the predictive power of different models and is able to predict gene conversion more accurately than any of the two studied models. Our work provides a useful framework for future improvement of gene conversion predictions through multiple models of gene conversion.

1 Introduction

Gene conversion is a process by which all or part of the sequence of a gene is changed to match the sequence of another gene. Figure 1 gives a schematic demonstration of how gene conversion between two sequences occurs. Its role in both short-term and long-term evolution of genes and genomes cannot be overemphasized. Gene conversion is an important process that can affect the evolutionary fate of duplicated genes. For instance, gene conversion can affect how similar two duplicated genes remain after gene duplication. After gene duplication, depending on which gene is the donor, any mutations in one gene could either be made to disappear or be transferred to the other copy through gene conversion. On the one hand, very frequent gene conversion can lead to two highly identical duplicated genes even long after their duplication. This supports the false impression that the two genes arose from a recent gene duplication as their sequences are highly similar. On the other hand, gene conversion can boost the amount of variation and divergence between duplicated genes, e.g., in the major histocompatibility complex (MHC) genes [1].

Detecting gene conversion is computationally challenging. A number of methods have been proposed for detecting recombinations, of which gene conversion is one type. However, to our knowledge, only a couple of programs

(e.g. GENECONV [2]) are specifically designed for detecting gene conversions. Moreover, previous comparisons of the power of existing software to test for recombination have indicated that, while some perform better than others, no single program is universally superior [3]. Similarly, little is known about which program is the best for gene conversion prediction. To address this problem and to take advantage of the existing programs of recombination prediction, we propose an ensemble based learning method to combine optimally the predictions from different programs to form a unified program prediction and to boost the power of gene conversion prediction.

As in real life, the incidence of gene conversion in a gene family may be restricted to only a few members, and thus the majority cases are gene pairs that do not exhibit gene conversions. Prediction of gene conversion thus falls into the rare-class prediction problem, which is a common phenomenon and has been an issue with many classification problems [4]. Unfortunately, the majority of classifiers will label all data objects as belonging to the majority class. Thus a high accuracy is achieved, however none of the rare-class data objects are correctly identified. There are a variety of ways to deal with rare-class datasets (a paper by Sun et al. provides a nice overview [5]) but for our purposes we went with an implementation of the Meta-Cost algorithm [6] as our cost-sensitive learner.

In this study, we investigated the use of cost-sensitive learning, to combine the output of two programs that are specifically designed for detecting gene conversion, GENECONV [2] and Partimatrix [7]. Our preliminary studies indicate that the cost-sensitive classifier performs better than either of the two programs.

2 Methods

A variety of programs exist that deal with gene conversions and recombination. However few programs exist that provide a definitive prediction on between which two sequences in a set of sequences a gene conversion occurred. Two such programs are GENECONV [2] and Partimatrix [7] which we elaborate next.

GENECONV was developed specifically to identify gene conversions. Given an aligned file of sequences, it will give a prediction of what sequence fragments have the highest, unique similarity between two sequences. It then presents p -values that show the significance of these paired sequence fragments. p -values are determined both globally and pairwise. Global p -values are determined by comparing each fragment with all other possible fragments, thus establishing the overall uniqueness of the paired fragments. Pairwise compares each fragment

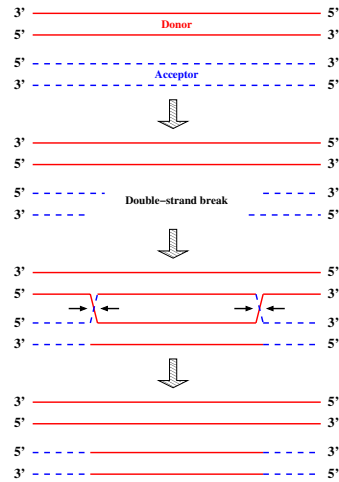


Fig. 1. Gene conversion between two sequences

with the maximum score that would be expected had gene conversion not occurred.

Partimatrix is designed to identify cases of gene conversion and recombination and analyze anomalous phylogenetic history in a multiple sequence alignment. For each sequence pair it determines a support score for whether a recombination has occurred and a conflict score that identifies the uniqueness of this recombination. If the support score is significantly higher than the conflict score, this represents evidence for a recombination event between these sequences.

The basic idea behind ensemble-based learning is to use multiple tools that predict, for example, gene conversion, and to use these tools in tandem to achieve prediction results better than each tool could produce alone. Neither GENECONV nor Partimatrix is 100% accurate when it comes to determining gene conversions and preliminary work we did with these programs showed that they would identify correct gene conversions at different times. After running extensive evaluations, we determined that GENECONV performs better in situations where the gene conversion event was recent and the sequences were not very diverged. Partimatrix on the other hand performs better on “ancient” duplication events where the sequences are also more diverged. So through combining them and also features that are measurable that may be associated with gene conversion, we hoped to create an ensemble classifier that would predict gene conversions more accurately than both of these programs.

Each row of attributes used in the classifier is based on data from a pair of sequences. This means that we must divide each set of sequences into a set of sequence pairs. So a set of six sequences for instance would constitute 15 sequence pairs and thus 15 rows of attributes. The attributes include the following data values: average GC content of both sequences, sequence identity between the sequences, global and pairwise *p*-values from GENECONV, and the support and conflict scores from Partimatrix.

Especially when considering each row of attributes as a pair of sequences, the likelihood of there being a gene conversion is very slim. Due to this fact, we looked into methods of creating a classifier that can handle learning a rare-class. For this we implemented a version of the MetaCost classifier first proposed by Domingos [6]. This method takes a cost matrix and applies it to a base classifier in order to adjust the weights of the training data.

Since our situation is a two-class problem (a gene pair is either labeled as gene conversion or no gene conversion), the cost matrix is a 2x2 matrix (as seen in Table 1). Each of the cells corresponds to the following values: true positives, false negatives, false positives, and true negatives. Within each of these cells, the

Table 1. Cost Matrix

True Positive (Correctly Identified Gene Conversions)	False Positive (Incorrectly Identified Gene Conversions)
False Negative (Incorrectly Identified No Gene Conversions)	True Negative (Correctly Identified No Gene Conversions)

user can then set the penalty for misclassifying a row or potentially the reward for correctly classifying one. This cost matrix is then applied to a base classifier.

The difficulty with this approach however is selecting the right cost matrix. In our situation, we definitely want to penalize false negatives (situations in which there was a gene conversion, yet the pair was classified as having none) and reward true positives. However, we do not want this to increase the amount of false positives too much as well, so a penalty must also exist for them. Finding this balance is certainly non-trivial and we have developed a method to do it automatically based on a greedy-search approach.

Starting with a basic, default cost matrix in which the penalty is 1 for both false positives and false negatives and 0 for both true positives and true negatives, we determine how the base classifier does after being trained on the training data and tested on an initial test set. We then increase the penalty of false negatives and false positives by 1 and decrease the reward value for true positives by 1 and every combination of these. We do not adjust the reward for true negatives as this represents the majority class (correctly identified pairs with no gene conversion). After seeing how the classifier performs on every combination of incrementing/decrementing the cost matrix values (7 in total), we then take the best combination and then proceed using that as the base matrix. The determination of best is done through the use of the F-measure ($= 2 / (1 / \text{Recall} + 1 / \text{Precision})$), which represents the harmonic mean between recall and precision. Because accuracy is not a good metric in rare-class predictions, a common approach is to use two different metrics: recall and precision [5,8,9]. Recall ($= \text{True Positives} / (\text{True Positives} + \text{False Negatives})$) “measures the fraction of positive examples correctly predicted by the classifier”, and precision ($= \text{True Positives} / (\text{True Positives} + \text{False Positives})$) “determines the fraction of [data objects] that actually turns out to be positive in the group the classifier has declared as a positive class” [10]. These are better metrics than accuracy as they indicate how many of the rare-class are identified while still indicating a balance between true and false positives. However, the disadvantage is that sometime it can be difficult to compare two metrics simulatenously and find a classifier that give the best values for both metrics. We therefore adopted the single metric–F-measure to evaluate different classifiers. The cost matrix that produces the best F-measure is used subsequently as the best cost matrix. Since not every iteration produces a best cost matrix that is better than the previous best cost matrix, we keep track of an overall best cost matrix. This can then be repeated for a user-specified number of iterations (we used 30).

We used four base classifiers as part of our experiments. The first is an implementation of the NaiveBayes learner. The second is J4.8 which is a variation of the C4.5 decision tree learner. The third is PART which is a rule-based learner that creates rules based on partial C4.5 decision trees [11]. The fourth is called JRip which is an implementation of the RIPPER algorithm [12]. We implemented these classifiers in the Weka learning environment [13].

As few examples of actual gene conversion events exist, we had to generate our own training data. The generation of the sequences was done in similar fashion

to that by Marais [14]. Basically a set of sequences was generated from a randomly generated phylogenetic tree, thus creating a simulated gene family. These sequences are mutated over time and a gene conversion event is inserted between two sequences. Since genes that exhibit gene conversion are known to have increased GC levels, we included a GC bias when inserting the sequence fragment into the acceptor gene. These sequences were aligned with MUSCLE [15].

We generated two separate training sets under different conditions. The first set (SET1) was done under conditions that mirror typical features of gene conversions [16]. In this case GENECONV performs very well under these conditions (and Partimatrix not so well), we created a second dataset (SET2). Recall, Partimatrix performs better in conditions where the sequences are highly diverged and the gene conversion event happened early (i.e. an ancient gene conversion event). This second dataset was created to reflect this situation. In SET1 the overall average sequence identity was 0.83 (with values ranging between 0.99 and 0.66) and in SET2 the overall average identity was 0.49 (with values ranging between 0.6 and 0.39). Each training dataset contained 500 sets of sequences. Each set of sequences contained 6 sequences, in which one gene conversion has taken place. For each training set, two additional test sets of 100 sequences were generated, one to determine the cost matrix and one to test the cost matrix.

3 Results

The results for the various classifiers can be seen for SET1 and SET2 in Table 2 and Table 3, respectively. The upper part represents the “basic classifiers” while the lower part represents the classifiers that were used as the base classifiers with the MetaCost implementation and the generated cost matrix. The basic classifiers are as follows: “Perfect” represents an ideal classifier and is included to see how each of the classifiers compare to it. “Just Say No” represents a classifier where every gene pair is considered to have no gene conversion. As can be seen here, a classifier like this has a relatively high accuracy, but a recall of 0 and an undefined precision (due to a division by zero) and F-measure. “GENECONV Strict” represents using GENECONV with its default settings and only evaluating the global p -values. “GENECONV LP” also uses the local pairwise p -values. Partimatrix does not actually give a finite prediction on whether a gene conversion has occurred so we took the pair that has the lowest conflict score as the pair that exhibits gene conversion. “G-or-P” represents a basic combining of the two classifiers based on one rule: if either GENECONV LP or Partimatrix says a gene conversion has occurred, then the pair is labeled as having a gene conversion. In addition, since Partimatrix does not perform as well on sequences that are too similar it gave no results for some pairs in SET1. These pairs were excluded and a total of 2190 pairs were used instead of 2250.

In SET1, GENECONV performs well. “GENECONV Strict” has a high accuracy, even higher than the “Just Say No approach”. However, through our method we are able to increase the amount of true positives, increase the accuracy, and most importantly, increase the F-measure. The best performers are

Table 2. SET1

Classifier	TP	FP	TN	FN	Accuracy	Recall	Precision	F-measure
<i>Perfect</i>	139	0	2051	0	1	1	1	1
<i>Just Say No</i>	0	0	2051	139	0.937	0	UNDEF	UNDEF
<i>GENECONV Strict</i>	102	4	1448	35	0.975	0.745	0.962	0.840
<i>GENECONV LP</i>	123	57	1395	14	0.955	0.898	0.683	0.776
<i>Partimatrix</i>	9	137	1315	128	0.833	0.066	0.062	0.064
<i>G-or-P</i>	128	191	1261	9	0.874	0.934	0.401	0.561
<i>NaiveBayes</i>	122	58	1394	15	0.954	0.891	0.678	0.770
<i>PART</i>	107	5	1447	30	0.978	0.781	0.955	0.859
<i>J4.8</i>	109	11	1441	28	0.975	0.796	0.908	0.848
<i>JRip</i>	111	9	1443	26	0.978	0.810	0.925	0.864

Table 3. SET2

Classifier	TP	FP	TN	FN	Accuracy	Recall	Precision	F-measure
<i>Perfect</i>	150	0	2250	0	1	1	1	1
<i>Just Say No</i>	0	0	2100	150	0.933	0	UNDEF	UNDEF
<i>GENECONV Strict</i>	1	8	2092	149	0.930	0.007	0.111	0.014
<i>GENECONV LP</i>	5	68	2032	145	0.905	0.033	0.068	0.045
<i>Partimatrix</i>	15	135	1965	135	0.880	0.100	0.100	0.100
<i>G-or-P</i>	19	197	1903	131	0.854	0.127	0.088	0.104
<i>NaiveBayes</i>	8	75	2025	142	0.904	0.053	0.096	0.069
<i>PART</i>	35	214	1886	115	0.854	0.233	0.141	0.175
<i>J4.8</i>	23	160	1940	127	0.872	0.153	0.126	0.138
<i>JRip</i>	40	265	1835	110	0.833	0.267	0.131	0.176

JRip and PART, which is not surprising as they are rule-based classifiers and rule-based classifiers are known to perform well on rare-class data [10]. Both have a higher F-measure than “GENECONV Strict”, a higher accuracy, and both identify more true positives. J4.8 does well too and identifies more true positives as PART, but more false positives as well. Of all the cost matrix classifiers, NaiveBayes identifies the most true positives, but is hindered by the amount of false positives it identifies.

In SET2, GENECONV performs poorly. Partimatrix identifies more gene conversions and G-or-P has the best F-measure of these basic classifiers. This set also shows the shortcoming of using accuracy as a metric as the “Just Say No” approach would appear to be the best classifier. Among the cost matrix classifiers, the NaiveBayes classifiers performs quite poorly. It has an F-measure lower than G-or-P, so it shows no improvement over a basic classifier (it does not identify more gene conversions correctly either). But the rule-based classifiers again perform quite well, with both identifying more gene conversions and having higher F-measures than any of the basic classifiers. In fact, aside from NaiveBayes, all classifiers exhibit both a higher recall and a higher precision than the basic classifiers, showing a definite improvement.

Table 4. Cost Matrices

NaiveBayes SET1	-3	2	NaiveBayes SET2	-2	1
	2	0		2	0
PART SET1	-4	5	PART SET2	-3	19
	3	0		1	0
J4.8 SET1	-2	4	J4.8 SET2	0	4
	3	0		1	0
JRip SET1	-4	6	JRip SET2	0	7
	1	0		1	0

Table 4 shows the cost matrices that were determined for each classifier by the greedy-based approach and subsequently used to make gene conversion predictions. It indicates that a cost matrix is highly dependent on both the classifier and the data being used. No classifier has the same cost matrix across both datasets and no dataset has a cost matrix that is best for more than one classifier. In fact, all cost matrices that were determined by our approach are unique.

Many trends are noticeable when looking at the cost matrices. If we exclude NaiveBayes, which performed poorly on both datasets, all classifiers have higher penalties for false negatives than false positives. This makes sense as the initial problem most classifiers have when dealing with rare-class problems is identifying those rare-class members and reducing the number of false negatives.

Further interesting to note are trends in the cost matrices for each classifier when comparing them across the two datasets. The reward for true positives for SET1 is generally “greater” (i.e., the lower the value is, the greater the reward) than that for SET2. This seems to reflect the relative comparison of the true positives between SET1 and SET2 as the number of true positives predicted by GENECONV and Partimatrix in SET1 are much higher than that in SET2 (see Tables 2 and 3). Moreover, that the penalty for false positives in SET2 is higher or equal to that in SET1 seems also to reflect the relative comparison of the number of false positives in SET1 and SET2; the number of false positive predictions in SET2 are slightly higher than that in SET1. Comparatively, the difference in the penalty for false negatives between SET1 and SET2 does not seem to be reflected by the difference in the number of false negative predictions between the two sets.

4 Discussion

In our analysis we have shown how a greedy-search for a best cost matrix can be a quite effective method to deal with a rare-class dataset and help with the identification of gene conversions. This method paired with a rule-based classifier seems to be an effective and robust pairing as in both datasets JRip and PART were the best classifiers (with JRip usually outperforming PART). Through our method we have not only improved upon each of the programs GENECONV

and Partimatrix but also on their basic combination (G-or-P). This represents an exciting step forward in the accurate prediction of gene conversions.

An interesting observation is the necessity for a way to generate a best cost matrix. As revealed by our analysis, a cost matrix that maximizes the F-measure is highly dependent on both the dataset and the classifier being used. No cost matrix was the same across datasets or classifiers. The majority of research on cost matrices deals with how to implement them [6] and not with how to generate a cost matrix. However, due to this dependency on both data and classifier, it is certainly not a trivial task and further research into generating an “ideal” cost matrix in an efficient and accurate manner is warranted.

SET1 represents a set of sequences in which the gene conversion occurred relatively recently in the life-cycle of the gene family whereas the sequences in SET2 had a gene conversion that was earlier and thus more obscured by mutations that have occurred in the sequences since. Our method shows more improvement in SET2 than it does in SET1. In SET1 the best improvement is approximately 0.024 better in terms of F-measure (JRip better than GENECONV Strict). But in SET2, we have an F-measure improvement of greater than 0.07 (JRip better than G-or-P). This shows that while our approach does improve prediction in both situations, it improves the identification of “older” gene conversions more than more recent ones. Future studies can be directed to better identify these relative ancient gene conversion by taking account of sequence mutation patterns and also incorporating additional gene conversion prediction models.

References

1. Yeager, M., Hughes, A.L.: Evolution of the mammalian mhc: Natural selection, recombination, and convergent evolution. *Immunological Reviews* 167, 45–58 (1999)
2. Sawyer, S.: Statistical tests for detecting gene conversion. *Molecular Biology and Evolution* 6(5), 526–538 (1989)
3. Posada, D., Crandall, K.A.: Evaluation of methods for detecting recombination from dna sequences: Computer simulations. *Proceedings of the National Academy of Sciences of the United States of America* 98(24), 13757–13762 (2001)
4. Chawla, N., Japkowicz, N., Kolcz, A.: Editorial: Special issue on learning from imbalanced datasets. *SIGKDD Explorations Special Issue on Learning from Imbalanced Datasets* (2004)
5. Sun, Y., Kamel, M.S., Wong, A.K.C., Wang, Y.: Cost-sensitive boosting for classification of imbalanced data. *Pattern Recognition* 40, 3358–3378 (2007)
6. Domingos, P.: Metacost: A general method for making classifiers cost-sensitive. In: *Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining (KDD 1999)* (1999)
7. Jakobsen, I.B., Wilson, S.R., Eastal, S.: The partition matrix: Exploring variable phylogenetic signals along nucleotide sequence alignments. *Molecular Biology and Evolution* 14(5), 474–484 (1997)
8. Joshi, M.V., Kumar, V., Agarwal, R.C.: Evaluating boosting algorithms to classify rare classes: comparison and improvements. In: *ICDM* (2001)
9. Davis, J., Goadrich, M.: The relationship between precision-recall and ROC curves. In: *Proceedings of the 23rd International Conference on Machine Learning* (2006)

10. Tan, P.N., Steinbach, M., Kumar, V.: *Introduction To Data Mining*. Addison-Wesley, Reading (2006)
11. Frank, E., Witten, I.H.: Generating accurate rules sets without global optimization. In: *Fifteenth International Conference on Machine Learning* (1998)
12. Cohen, W.W.: Fast effective rule induction. In: *Machine Learning: Proceedings of the Twelfth International Conference (ML 1995)* (1995)
13. Witten, I.H., Frank, E.: *Data Mining: Practical machine learning tools and techniques*, 2nd edn. Elsevier, Amsterdam (2005)
14. Marais, G.: Biased gene conversion: implications for genome and sex evolution. *Trends Genet.* 19(6), 330–338 (2003)
15. Edgar, R.C.: Muscle: A multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics* 5, 1–19 (2004)
16. Chen, J.M., Cooper, D.N., Chuzhanova, N., Ferec, C., Patrinos, G.P.: Gene conversion: mechanisms, evolution and human disease. *Nature Reviews Genetics* (2007)

A Multi-objective Genetic Algorithm Based Approach to the Optimization of Oligonucleotide Microarray Production Process

Filippo Menolascina¹, Vitoantonio Bevilacqua¹, Caterina Ciminelli¹,
Mario Nicola Armenise¹, and Giuseppe Mastronardi¹

Department of Electrotechnics and Electronics, Technical University of Bari,
70126, Italy
f.menolascina@ieee.org

Abstract. Microarrays are becoming more and more utilized in the experimental platform in molecular biology. Although rapidly becoming affordable, these micro devices still have quite high production cost which limits their commercial appeal. Here we present a novel multiobjective evolutionary approach to the optimization of the production process of microarray devices mainly aimed at lowering the number of fabrication steps. In order to allow the reader to better understand what we describe we report herein a detailed description of a real-world study case carried out on the most recent microarray platforms of the market leader in this field. A comparative analysis of the most widely used approaches, main potentialities and drawbacks of the proposed approach are presented.

1 Introduction

An oligonucleotide microarray is a piece of glass or plastic material on which single-stranded fragments of DNA, called probes, are placed or synthesized. The chips produced, for instance, can contain more than one million spots (or features) as small as $11 \mu\text{m}$, with each spot accommodating several million copies of a probe. Probes are typically 25 nucleotides long and are synthesized in parallel, on the chip, in a series of repetitive steps. Each step appends the same nucleotide to probes of selected regions of the chip. Selection occurs by exposure to light with the help of a photolithographic mask [1].

Formally, we have a set of probes $P = \{p_1, p_2, \dots, p_n\}$ that are produced by a series of masks $M = \{m_1, m_2, \dots, m_T\}$, each mask m_t allowing the addition of a particular nucleotide $S_t \in \{A, C, G, T\}$ to be included in a subset of P . The nucleotide deposition sequence $S = S_1 S_2 \dots S_T$ corresponding to the sequence of nucleotides added at each masking step is therefore a supersequence of all $p \in P$ [10].

In general, a probe can be embedded within S in several ways. The embedding step of p_k can be described as T -tuple $\epsilon_k = (e_{k,1}, e_{k,2}, \dots, e_{k,T})$ in which $e_{k,t} = 1$ if probe p_k receives nucleotide S_t (at step t), or 0 otherwise. The deposition sequence is often denoted repeated permutation of the alphabet, mainly because

of its regular structure and because such sequences maximize the number of distinct subsequences. We distinguish between *synchronous* and *asynchronous* embeddings. In the first case, each probe has exactly one nucleotide synthesized in every cycle of the deposition sequence; hence, 25 cycles or 100 steps are needed to synthesize probes of length 25. In the case of asynchronous embeddings, probes can have any number of nucleotides synthesized in any given cycle, allowing shorter deposition sequences. All chips manufactured by this producer of can be asynchronously synthesized in 74 steps (18.5 cycles), which is probably due to careful probe selection. The problem of finding the sequence that reduces the number of steps required to accomplish the microarray production process is called SCS (Short Common Supersequence). The SCS problem is well-known to be NP-complete. In this paper, we present a novel approach to the problem of finding the SCS based on a multi-objective genetic algorithm that tries to minimize both the number of steps and the number of mask change in order to minimize the costs related to microarray production.

2 The SCS Problem with Applications in Bioinformatics and Nanotechnology

2.1 Microarray Techonology

Several microarray technologies are available today, based on a variety of fabrication techniques including printing with fine-pointed pins onto glass slides, ink-jet printing, electrochemistry on microelectrode arrays and photolithography. This paper is mainly concerned with the production of high-density oligonucleotide microarray, also called DNA chips or gene chips, that are fabricated by photolithography. This type of microarray consists of relatively short DNA probes synthesized at specific locations, named features or spots, on a solid surface. Each probe is a single-stranded DNA molecule of 10 to 70 nucleotides that perfectly matches with a specific portion of a target molecule. sequence of nucleotides added in each step is called deposition sequence or synthesis schedule. The selection of which probes receive the nucleotide is achieved by photolithography [1][2]. Figure 1 illustrates this process: The quartz wafer of a GeneChip array is initially coated with a chemical compound topped with a light-sensitive protecting group that is removed when exposed to ultraviolet light, activating the compound for chemical coupling. A lithographic mask is used to direct light and remove the protecting groups of only those positions that should receive the nucleotide of a particular synthesis step. A solution containing adenine (A), thymine (T), cytosine (C) or guanine (G) is then flushed over the chip surface, but the chemical coupling occurs only in those positions that have been previously deprotected. Each coupled nucleotide also bears another protecting group so that the process can be repeated until all probes have been fully synthesized.

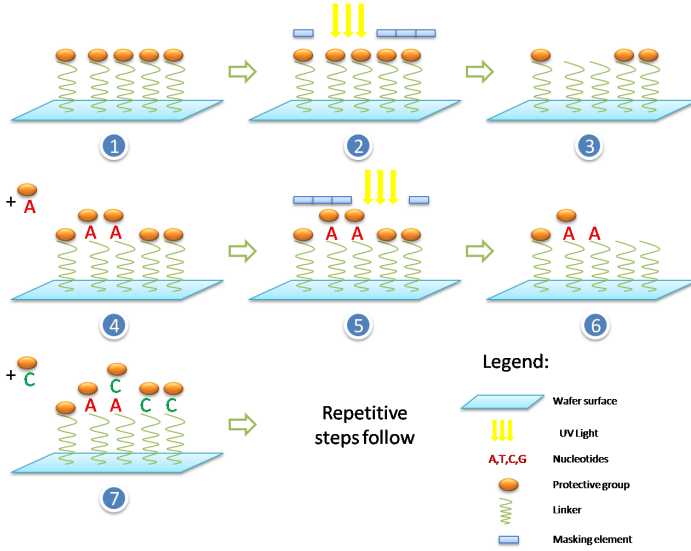


Fig. 1. Probe synthesis via photolithographic masks. The chip is coated with a chemical compound and a light-sensitive protecting group; masks are used to direct light and activate selected probes for chemical coupling; nucleotides are appended to deprotected probes; the process is repeated until all probes have been fully synthesized.

2.2 SCS Problem

The problem of finding the Shortest Common Supersequence (SCS) of a given set of sequences is a very important problem in computer science, especially in computational molecular biology. The SCS of a set of sequences can be stated as follows: Given two sequences $S = s_1s_2 \dots s_m$ and $T = t_1t_2 \dots t_n$, over an alphabet set $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_n\}$, we say that S is the subsequence of T (and equivalently, T is the *supersequence* of S) if for every s_j , there is $s_j = t_{i_j}$ for some $1 \leq i_1 < i_2 < \dots < i_m \leq n$. Given a finite set of sequences $S = \{S_1, S_2, \dots, S_k\}$, a common supersequence of S is a sequence T such that T is a supersequence of every sequence $S_j (1 \leq j \leq k)$ in S . Then, a shortest common supersequence (SCS) of S is a supersequence of S that has minimum length. In this paper, we shall assume that k is the number of sequences in S , n is the length of each sequence, and $q = |\Sigma|$ is the size of the alphabet. The SCS problem has applications in many diverse areas, including data compression [3], scheduling [4], query optimization [5], text comparison and analysis, and biological sequence comparisons and analysis [6][7]. As a result, the SCS problem has been very intensively investigated [8][9]. One basic result is that the SCS of two sequences of length n can be computed using dynamic programming in $O(n^2)$ time and $O(n^2)$ space (see, for example, [10]). There are also several papers that reported improvements on the running time and space required for dynamic programming algorithms [9]. For a fixed k , the dynamic programming algorithm can be extended to solve the SCS problem for k sequences of length n in $O(n^k)$ time

and space. Clearly, this algorithm is not practical for large k . The general SCS problem on arbitrary k sequences of length n is well-known to be NP-hard. In fact, Jiang and Li [10] showed that even the problem of finding a constant ratio approximation solution is also NP-hard.

Previous Research in SCS problem. We now present a brief survey of the most popular heuristic algorithms proposed in literature. Let S be any instance of the SCS problem and let $CS_A(S)$ be the supersequence of S identified by a heuristic algorithm A . Let $opt(S)$ denote an optimal solution for the instance S . Then, we say that A has an approximation ratio of λ if $|CS_A(S)|/|opt(S)| \leq \lambda$ for all instances S .

Alphabet Algorithm

The Alphabet algorithm is a quite simple approach to the problem under investigation [8]. Let S be a set of sequences of maximum length n over the alphabet $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_q\}$, then the Alphabet algorithm outputs a common supersequence of $(\sigma_1, \sigma_2, \dots, \sigma_q)^n$. The Alphabet algorithm has an approximation ratio of $q = |\Sigma|$. The time complexity of the Alphabet algorithm is $O(qn)$. There have also been modifications of the Alphabet algorithm that uses information from S to ‘remove’ redundant characters in $(\sigma_1, \sigma_2, \dots, \sigma_q)^n$. These methods improve the performance in practice, but not in the worst case approximation ratio of q .

Majority Merge Algorithm

The Majority-Merge algorithm [10] (MM) is a simple, greedy heuristic algorithm. Let’s suppose we analyze every sequence from left to right, the frontier is defined as the rightmost characters to be analyzed. Initially, the supersequence CS is empty. At each step, let s be the majority among the ‘frontier’ characters of the remaining portions of the sequences in S . Set $CS = CS||s$ (where $||$ represent concatenation) and delete the ‘frontier’ s characters from sequences in S . Repeat until no sequences are left. This algorithm is the same as the Sum Height algorithm (SH) proposed in [12]. This algorithm does not have any worstcase approximation ratio, but performs very well in practice. The time complexity of the Majority-Merge algorithm is $O(qkn)$.

Greedy and Tournament algorithms

The Greedy algorithm (GRDY) and Tournament algorithm (TOUR) studied in [13] are two variations of an iterative scheme based on combining optimal sequence pairs. Given any pair of sequences, S_i and S_j , an optimal supersequence of the pair, denoted by $SCS(S_i, S_j)$, can be computed in $O(n^2)$ using dynamic programming. The Greedy algorithm first chooses the ‘best’ sequence pair the that gives the shortest $SCS(S_i, S_j)$. Without loss of generality, we assume that these two sequences are S_1 and S_2 . The algorithm then replaces the two sequences S_1 and S_2 by their supersequence, $SCS(S_1, S_2)$. The algorithm proceeds recursively. Thus, we can express it as follows:

$$\text{Greedy}(S_1, S_2, \dots, S_k) = \text{Greedy}(SCS(S_1, S_2), S_3, \dots, S_k)$$

The Tournament algorithm is similar to the Greedy algorithm. It builds a ‘tournament’ based on finding multiple best pairs at each round and can be expressed schematically as follows:

$$\text{Tournament}(S_1, S_2, \dots, S_k) = \text{Tournament}(SCS(S_1, S_2), SCS(S_3, S_4), \dots, SCS(S_{k-1}, S_k)).$$

Both Greedy and Tournament algorithms have $O(k^2n^2)$ time complexity and $O(kn + n^2)$ space complexity. Unfortunately, it was shown in [11] that both Greedy and Tournament do not have approximation ratios.

3 Multi-objective Genetic Algorithms in Microarray Production Process Optimization

Multi-Objective Genetic Algorithms (MOGAs) are a relatively recent extension of Genetic Algorithms (GAs) that are well established bio-inspired computational optimization approaches with a wide range of applications that spans from finance to medicine. The concept of GA was developed by Holland and his colleagues in the 1960s and 1970s [14]. GA are inspired by the evolutionist theory explaining the origin of species. In nature, weak and unfit species within their environment are faced with extinction by natural selection. The strong ones have greater opportunity to pass their genes to future generations via reproduction. If these changes provide additional advantages in the challenge for survival, new species evolve from the old ones. Unsuccessful changes are eliminated by natural selection. In GA terminology, a solution vector $x \in X$ is called an *individual* or a *chromosome*. Chromosomes are made of discrete units called *genes*. Each *gene* controls one or more features of the chromosome. In the original implementation of GA by Holland, genes are assumed to be binary digits. In later implementations, more varied gene types have been introduced. Normally, a chromosome corresponds to a unique solution x in the solution space. This requires a mapping mechanism between the solution space and the chromosomes.

Being a population-based approach, GA are well suited to solve multi-objective optimization problems. A generic single-objective GA can be modified to find a set of multiple non-dominated solutions in a single run. The ability of GA to simultaneously search different regions of a solution space makes it possible to find a different set of solutions for difficult problems with non-convex, discontinuous, and multi-modal solutions spaces. Most multi-objective GA do not require the user to prioritize, scale, or weigh objectives. Therefore, GA have been the most popular heuristic approach to multi-objective design and optimization problems. Jones et al. [15] reported that 90% of the approaches to multiobjective optimization aimed to approximate the true Pareto front for the underlying problem. A majority of these used a meta-heuristic technique, and 70% of all metaheuristics approaches were based on evolutionary approaches. From this perspective it

could be easily intended how MOGAs can be used in order to carry out an optimization that aims at pursuing a minimization in terms of the number of steps required for manufacturing a microarray and, contemporary, to minimize set up times costs associated to the change in base to base to be deposited on the surface of the so far assembled biochip. In the next paragraph we will show how this problem has been addressed using a MOGA and we will expose computational results relating to a specific instance of this problem.

4 Novel Optimisation Procedure

In the proposed experimental design we evaluated the performance of the MOGA approach (Pareto front in Fig. 2) i.e. the times required. In order to make our evaluation as close to a real case as possible we used 4 oligonucleotide sequences taken from the most advanced chip for gene expression evaluation from the market leader. It is well known that the manufacturing process consists of depositing nucleotides in a step-by-step flavor, so as to reach the 25 oligonucleotide length for each of the features on the array. We selected the sequences:

```
8146645 gaagactcgctgtgggacagcgc
8054479 gcatgtggctacttagtaaatagta
8154660 gcttagaaaacaggtcctcagcaca
8162631 ggtagcaaccgtcacaatctggatg
```

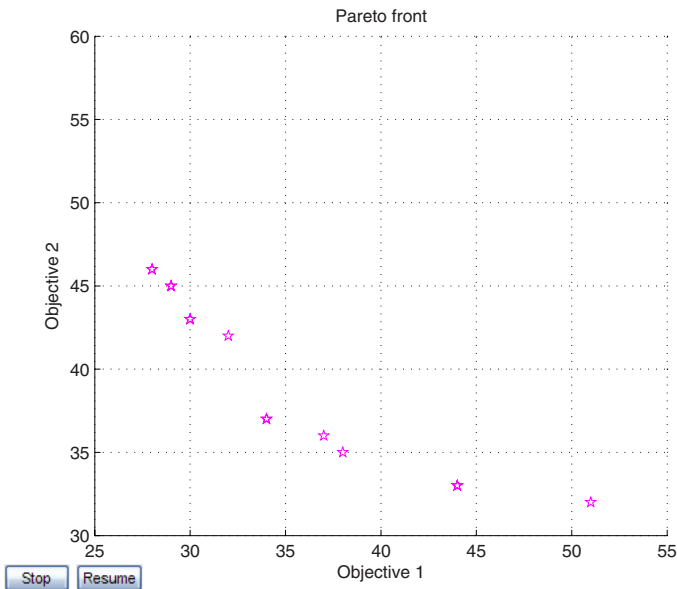


Fig. 2. Pareto front of the solutions found by the MOGA

Table 1. The deposition sequence and the corresponding embedding matrix

	G G C T A C G T G C T G A C A G C T A G C G A
	1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
ATCGAAGCGCGA	- - - - A - - T - C - G A - A G C - - G C G A
CCGTCCAGCTAA	- - C - C G T - C - - - - C A G C T A - A - -
GTACTTAGCTAC	G - - T A C - T - - T - A - G C T A - C - - -
GGATGGAGCTAC	G G - - A - - T G - - G A - - G C T A - C - -
Embedding Matrix	- - - - 1 - - 1 - 1 - 1 1 - 1 1 1 - 1 1 1 1
	- - 1 - 1 1 1 - 1 - - - - 1 1 1 1 1 1 - 1 - -
	1 - - 1 1 1 - 1 - - 1 - 1 - 1 1 1 1 - 1 - - -
	1 1 - - 1 - - 1 1 - - 1 1 - - 1 1 1 1 - 1 - -

and we evaluated the performances of the algorithm using the following protocol: we started taking 12 bases from each of sequence and running the optimization algorithm 100 times on the same problem. We proceeded by adding each time one base to the previous sequence and re-evaluating the performances of the algorithm, until the end of the sequences has been reached. At each step we recorded the time required for each optimization task, the number of steps required by each solution and the reason why the optimization algorithm ended. For each of the 14 tests carried out we extracted the mean and the confidence intervals for both optimization times and optimization results; this was done in order to extract main estimators of the performances of the proposed algorithm under the hypothesis that the process under observation is ergodic (so that the mean estimation is independent on the specific realization and that it tends to the real value for n , number of observations, $n \rightarrow \infty$). Computational time analysis of the optimization task revealed that times required by the proposed algorithm can be adequately fitted with a relatively low order polynomial that seems to enforce the thesis that states that the computational complexity of this approach can be approximated to $O(x^6)$. This algorithm was able to complete the optimization task proposed in the previous section using only 23 deposition setps are reported in Tab. 1. As it can be observed no deposition step can be suppressed without affecting the whole process. This suggest that the necessary condition for optimality is at least satisfied. The results reported herein seem to confirm the robustness and versatility of the proposed algorithm and push the need for further research in this field.

5 Discussion

In this paper, we have proposed a novel Multi-Objective approach for reduction of manufacturing steps required for microarray assembly. The algorithm is built on a MOGA that firstly generates random templates, and the Evolution process to reduce templates from template pool to get shorter and less expensive result. These processes are shown to be powerful for solving the SCS problem. Comparing the performance of our approach with the industry gold standard we can state

that the proposed system is able to outperform alternative approaches under pre-defined conditions. The proposed solution results to be quite interesting in terms of result optimality; however it should be noticed that computational complexity of the algorithm under investigation is not negligible and it requires many optimization tasks before completion. However much research effort must be spent on border conflicts minimization in microarray production due to the specific technological limitations that characterize the photolithographic processes.

References

1. Fodor, S., Read, J., Pirrung, M., Stryer, L., Lu, A., Solas, D.: Light-directed, Spatially Addressable Parallel Chemical Synthesis. *Science* 251, 767–773 (1991)
2. Hannenhalli, S., Hubell, E., Lipshutz, R., Pevzner, P.A.: Combinatorial Algorithms for Design of DNA Arrays. *Advances in Biochemical Engineering Biotechnology* 77, 1–9 (2002)
3. Storer, J.A.: *Data Compression: Methods and Theory*. Computer Science Press (1988)
4. Foulser, D.E., Li, M., Yang, Q.: Theory and Algorithms for Plan Merging. *Artificial Intelligence* 57(2), 143–181 (1992)
5. Sellis, T.K.: Multiple-query Optimization. *ACM Transactions on Database Systems (TODS)* 13(1), 23–52 (1988)
6. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: *Introduction to Algorithms*, 2nd edn. MIT Press/McGraw-Hill (2001)
7. Sankoff, D., Kruskal, J.: *Time Warps, String Edits and Macromolecules: the Theory and Practice of Sequence Comparisons*. Addison Wesley, Reading (1983)
8. Barone, P., Bonizzoni, P., Vedova, G.D., Mauri, G.: An Approximation Algorithm for the Shortest Common Supersequence Problem: an Experimental Analysis. In: *Symposium on Applied Computing, Proceedings of the 2001 ACM symposium on Applied computing*, pp. 56–60 (2001)
9. Gusfield, D.: *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*. Cambridge University Press, New York (1997)
10. Jiang, T., Li, M.: On the Approximation of Shortest Common Supersequences and Longest Common Subsequences. *SIAM Journal of Computing* 24(5), 1122–1139 (1995)
11. Timkovsky, V.G.: On the Approximation of Shortest Common Non-subsequences and Supersequences. Technical report (1993)
12. Kasif, S., Weng, Z., Derti, A., Beigel, R., DeLisi, C.: A Computational Framework for Optimal Masking in the Synthesis of Oligonucleotide Microarrays. *Nucleic Acids Research* 30(20) (2002)
13. Irving, R.W., Fraser, C.: On the Worst-Case Behaviour of Some Approximation Algorithms for the Shortest Common Supersequence of k Strings. In: *Proceedings of the 4th Annual Symposium on Combinatorial Pattern Matching*, pp. 63–73 (1993)
14. Holland, J.H.: *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor (1975)
15. Jones, D.F., Mirrazavi, S.K., Tamiz, M.: Multiobjective Meta-heuristics: An Overview of the Current State-of-the-art. *Eur. J. Oper. Res.* 137(1), 1 (2002)

A New Orthogonal Discriminant Projection Based Prediction Method for Bioinformatic Data

Chao Wang^{1,2,*} and Bo Li^{1,2}

¹ Intelligent Computing Lab, Institute of Intelligent Machine, Chinese Academy of Science,
P.O. Box 1130, Hefei, Anhui, 230031, China

² Departments of Automation, University of Science and Technology of China, Hefei, Anhui,
230027, China

wangchaopike@yahoo.com.cn

Abstract. DNA microarray allows the measurement of transcript abundances for thousands of genes in parallel. Though, it is an important procedure to select informative genes related to tumor from those gene expression profiles (GEP) because of its characteristics such as high dimensionality, small sample set and many noises. In this paper we proposed a novel method for feature extraction that is named as Orthogonal Discriminant Projection (ODP). This method is a linear approximation base on manifold learning approach. The ODP method characterizes the local and non-local information of manifold distributed data and explores an optimum subspace which can maximize the difference between non-local scatter and the local scatter. Moreover, it introduces the class information to enhance the recognition ability. A trick has been employed to handle the Small Sample Site (SSS). Experimental results on Non-small Cell Lung Cancer (NSCLC) and glioma dataset validates its efficiency compared to other widely used dimensionality reduction methods such as Principle Component Analysis (PCA), Linear Discriminant Analysis (LDA).

Keywords: NSCLC; SRBCT; Feature Extraction; GEP; ODP.

1 Introduction

Medical genetics seeks to understand how genetic variation relates to human health and disease [1]. When searching for an unknown gene that may be involved in a disease, researchers commonly use genetic linkage and genetic pedigree charts to find the location on the genome associated with the disease [2]. Although it is not an inherited disease, cancer is also considered a genetic disease.

Cancer classification is the critical basis for clinical therapy. With the development of technology, gene expression profiles (GEP) can be set up. The diagnosis based on GEP offers a systematic and precise prospect for tumor classification. DNA microarray technology is the most widely used method in cancer diagnosis, which has become a powerful tool in functional genome studies [3]. In the literature of microarray the “class

* Corresponding author.

prediction” has been named for this technique of cancer diagnosis based on the gene expression profiles.

There are many feature selection method focus on cancer classification. In 2000, Terrence S. Furey et.a offered a Support Vector Machines method for classification and validation between normal and cancer tissues [4]. In 2002, A. Antoniadis proposed dimensional reduction statistical techniques in conjunction with nonparametric discriminant procedures method [5]. In 2006, Huang and Zheng proposed Independent Component Analysis method for cancer data dimensional reduction [6]. In 2007, Liao and Chin proposed the logical regression method for disease classification in application of microarray data [7]. Also a comparative study of several discrimination methods based on filtered sets of genes has been proposed by Dudoit et al at 2002 [8]. Recently, a novel method based on manifold has been proposed for feature extraction from the original gene space.

In recent years, manifold learning based methods are becoming the most promising feature extraction approaches. Among them, one representative is Laplacian Eigenmap (LE) [9], which is also a spectral mapping method. LPP [10,11] is a linear approximation of LE. Inspired by the idea, J. Yang et al. presented a UDP algorithm [12]. However, it must be noted that both LPP and UDP are a linear approximation to the manifold learning approaches without taking the class information into account.

In this paper, Orthogonal Discriminant Projection (ODP), is proposed to overcome the problems mentioned above. Unlike the original manifold learning methods, ODP follows the supervised techniques and the label information is taken to model the manifold. Furthermore, combined to labels, both local information and non-local information is utilized to define the weights of any two points, which can explore the intrinsic structure of original data and enhance the recognition ability.

The paper is organized as follows: in Section 2.1, the PCA [15] algorithm is reviewed briefly. Then the Linear Discrimination Analysis (LDA) algorithm is reviewed in Section 2.2. In Section 3, the proposed method Orthogonal Discriminate Projection (ODP) is illuminated exactly. Some experiments are performed and comparisons with this method are also presented in Section 4.

2 Review of PCA and LDA

2.1 PCA

PCA seeks to find a subspace whose basis vectors correspond to the maximum-variance directions in the original space. Such that the global scatter is maximized after the projection of samples. The global scatter can be characterized by the mean square of the Euclidean distance between any pair of the projected sample points. Specifically, a set of training samples X_1, X_2, \dots, X_M of R^n dimensions. We get their images y_1, y_2, \dots, y_M after projecting the original points into a subspace.

The object function of PCA is defined by

$$J_T(w) = \max \frac{1}{2} \frac{1}{M} \frac{1}{M} \sum_{i=1}^M \sum_{j=1}^M \|y_i - y_j\|^2. \quad (1)$$

It follows that $y_i = w^T x_i \quad i = 1, \dots, M$. After deduction Eqn. (1) can be written as

$$J_T(w) = \max_{tr} \{w^T S_T w\} \tag{2}$$

where S_T is defined as follows:

$$S_T = \frac{1}{2} \frac{1}{M} \frac{1}{M} \sum_{i=1}^M \sum_{j=1}^M (x_i - x_j)(x_i - x_j)^T \tag{3}$$

Eqn.(3) indicate that S_T is essentially the covariance matrix of data. Applied the Lagrange multiplier to Eqn.(2). The optimized solution is $S_T w_i = \lambda w_i$. So the projection axes w that maximizing Eqn. (2) can be selected as the eigenvectors of the largest eigenvalues of S_T . Similarly, we can select a set of axes of PCA by selecting the d largest eigenvectors of S_T corresponding to the d largest eigenvalues.

2.2 LDA

LDA tries to find the optimal subspace that can maximize the between class scatter and minimize the within class scatter simultaneously. The fisher criterion is defined by:

$$J_F(w) = \max \frac{\|w^T S_B w\|}{\|w^T S_W w\|} \tag{4}$$

where S_B is the between-class and S_W is within-class scatter matrices. Obviously, that S_B and S_W are both semi-positive definite matrix and satisfy $S_B + S_W = S_T$. The optimized solutions of Eqn. (4) are the generalized eigenvectors w_1, w_2, \dots, w_d of $S_B w = \lambda S_W w$ corresponding to the d largest eigenvalues. These generalize eigenvectors form the coordinate system of LDA.

3 Orthogonal Discriminant Projection

In the proposed algorithm, we use difference instead of ratio between non-local scatter and local scatter as the definitions of weights, the local scatters and the non-local scatters are provided in the following subsections.

3.1 The Definition of the Neighbors

In manifold learning the adjacency matrix H among instances with the size of $M \times M$ should be defined. M represents the number of the total instances. Specifically, two samples X_i and X_j are viewed as a local σ -neighborhood if and only if $\|X_i - X_j\|^2 \leq \sigma$. The elements of the adjacency matrix H are given below:

$$H_{i,j} = \begin{cases} 1 & \text{if } \|X_i - X_j\|^2 \leq \sigma \\ 0 & \text{else} \end{cases} \tag{5}$$

Where k -neighborhood matrix H can be defined if X_j belong to the k nearest neighbors of instance X_i the parameter $H_{i,j} = 1$. The elements of the adjacency matrix H can be denoted as:

$$H_{i,j} = \begin{cases} 1 & \text{if } X_j \text{ is belong to } k \text{ nearest neighbors of } X_i, \\ 0 & \text{else.} \end{cases} \quad (6)$$

3.2 The Weights Matrix among Instances

In the original LE [9] or LPP [13], the weight between two instances is defined to be a heat kernel or simply either 1 or 0, which can not reflect their class information. In the proposed algorithm, the weights between any two points are defined based on both their local information and class information. The definition is stated as below.

$$W_{ij} = \begin{cases} L1 = \exp(-\frac{\|X_i - X_j\|^2}{\beta})(1 - \exp(-\frac{\|X_i - X_j\|^2}{\beta})) & \text{if } X_i \text{ and } X_j \text{ have the same label also among nearest neighbors} \\ L2 = 1 - \exp(-\frac{\|X_i - X_j\|^2}{\beta}) & \text{if } X_i \text{ and } X_j \text{ are among nearest neighbors with different labels} \\ L3 = 1 & \text{others} \end{cases} \quad (7)$$

where β is a parameter that is used as a regulator. We know that the magnitude of β will be the same size as the stand deviation of the data set.

3.3 The Local Scatter and No-Local Scatter

First, the adjacency matrix H takes the second definition of k -neighborhood. Then the local scatter has the following form:

$$\begin{aligned} J'_L(A) &= \frac{1}{2} \frac{1}{M} \frac{1}{M} \sum_{i=1}^M \sum_{j=1}^M H_{ij} W_{ij} (Y_i - Y_j)^2 \\ &= \frac{1}{2} \frac{1}{M} \frac{1}{M} A^T \sum_{i=1}^M \sum_{j=1}^M H_{ij} W_{ij} (X_i - X_j)(X_i - X_j)^T A \\ &= A^T S'_L A \end{aligned} \quad (8)$$

And local scatter matrix S'_L can be computed according to (9)

$$S'_L = \frac{1}{2} \frac{1}{M} \frac{1}{M} \sum_{i=1}^M \sum_{j=1}^M H_{ij} W_{ij} (X_i - X_j)(X_i - X_j)^T = \frac{1}{2} \frac{1}{M} \frac{1}{M} X L X^T \quad (9)$$

Where we define $Q_{ij} = H_{ij} W_{ij}$, then $L' = D' - Q$, $D' = \sum_j Q_{ij}$.

Based on the definition of local scatter, the non-local scatter can be computed from the following expression:

$$J'_N(A) = \frac{1}{2} \frac{1}{M} \frac{1}{M} \sum_{i=1}^M \sum_{j=1}^M (1 - H_{ij} W_{ij}) (Y_i - Y_j)^2 = A^T (S_T - S'_L) A \tag{10}$$

3.4 Justification

After the local scatter and the non-local scatter have been constructed, an optimization objective function can be devised to maximize the difference between the non-local scatter and the local scatter. That is:

$$J'(A) = J'_N(A) - J'_L(A) = A^T (S'_N - S'_L) A = A^T (S_T - 2S'_L) A \tag{11}$$

Also we can get the weight difference under constraint of $A^T A = I$

$$\arg \max_{A^T A = I} J'(A) = \arg \max_{A^T A = I} A^T (S_T - 2S'_L) A \tag{12}$$

We can easily find that A consists of the eigenvectors associated with d top eigenvalues of the following eigen-equation.

$$(S_T - 2S'_L) A_i = \lambda_i A_i \tag{13}$$

Thus, ODP can be applied to extract the most useful features from real-word data. The outline of the proposed algorithm is summarized as follows. Firstly, the weights between any two points constructed based on Eqn. (7). Secondly, the local scatter and the non-local scatter should be computed according to Eqn. (9) and Eqn. (10). Thirdly, obtain the optimized subspace A by solving Eqn. (13). At last, project a point to the low dimensional space using transform matrix A and predict its class label with a given classifier.

4 Experiments

Experiment A

This experiment is based on the Non-Small Cell Lung Cancer, in which the medical data of 174 patients with non-small cell lung cancer have been gathered in the early stage which are classified as stage I and stage II. The 37 features represent different conditions of the patients. If the life span longer than 60 months mean survival. In the experiment a binary prediction of the survival rate method has been set as follows, where those survived more than five years are labeled '0' and other cases labeled '1'. The first 80 patients were chosen as training and the other 94 as the test data. At the same time the CPU time is also listed for comparing. KNN are chose as the classifier in the experiment and K is set to 1.

Table 1 lists the prediction results based on the methods PCA, LDA and ODP. Every experiment has been done ten times and the average prediction rate is shown. The average CPU time is listed in Table 1 as well.

Table 1. The prediction rate of the methods

	Death	Live	Time
PCA	25.93%	86.57%	0.2882
LDA	48.15%	47.76%	0.8467
ODP	59.26%	65.67%	0.4108

Experiment results shows that, the elapse time LDA is four times of PCA but the prediction rate is awfully worse than PCA. The die prediction rate gives rise to 33.33% compared to PCA, the live prediction rate gives rise to 17.91% compared to LDA.

Experiment B

This experiment based on the glioma dataset. Glioma is a type of cancer that starts in the brain or spine of the central nervous system. The glioma cancer is classified by cell type, grade, and location. In this experiment we classify the dataset by the Pathology. Table 2 list some parts of the cancer data profile. In this experiment the training data set has 21 instances and the test set has 29 instances.

Table 2. Summary of clinical parameters for the high-grade glioma dataset

Sample Name	Pathology		Vital status	Survival (days)
Brain_CG_1	Classic	GBMa	Dead	308
Brain_CG_2	Classic	GBM	Dead	281
Brain_CG_3	Classic	GBM	Dead	501
Brain_CG_4	Classic	GBM	Dead	670
Brain_CG_5	Classic	GBM	Alive	729
Brain_CG_6	Classic	GBM	Dead	21

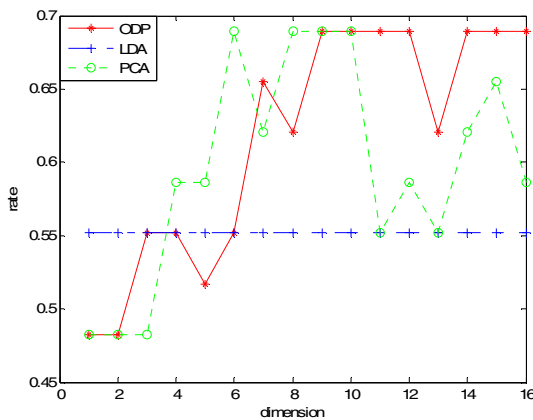


Fig. 1. The prediction rate of glioma

The prediction rates are shown in Fig.1. From this picture we can find that the new algorithm ODP could give out a stable prediction rate. The maximum prediction rate can keep steel on more than seven dimensions variance. While PCA only on 5, 8, 9, 10 dimensions have the maximum prediction rate. The most stable algorithm is LDA, but its prediction rate is low as the fig shows.

Table 3 give out the CPU time elapsed in the 16 time of experiment with the units second. In this experiment we also find the CPU time LDA is almost three times of ODP. While PCA is almost the same as the proposed algorithm when computing time consume.

Table 3. The CPU Time of three methods

	ODP	LDA	PCA
CPU Time	1.265	3.125	1.086

While β has the same amount as the stand deviation of the data set a good prediction rate will be reached. In the further research we will make the algorithm much easier to use and find the optimum β automatically.

Acknowledgments. This work was supported by the grants of the National Science Foundation of China, No. 30700161, the grant of the Guide Project of Innovative Base of Chinese Academy of Sciences (CAS), No.KSCX1-YW-R-30, and the grant of Oversea Outstanding Scholars Fund of CAS, No.2005-1-18.

References

1. Genes and Disease. NIH: National Center for Biotechnology Information. NCBI (2008)
2. Smith, D.G., Ebrahim, S.: Mendelian randomization: Can Genetic Epidemiology Contribute to Understanding Environmental Determinants of Disease? *International Journal of Epidemiology* 32, 1–22 (2003)
3. Alon, U., Barkai, N., Notterman, D.A., Gish, K., Ybarra, S., Mack, D., Levine, A.J.: Broad Patterns of Gene Expression Revealed by Clustering Analysis of Tumor and Normal Colon Tissues Probed by Oligonucleotide Arrays. *Proc. Natl. Acad. Sci. USA* 96, 6745–6750 (1999)
4. Furey, T.S., Cristianini, N., Duffy, N., David, W., Bednarski, Schummer, M., Haussler, D.: Support Vector Machine Classification and Validation of Cancer Tissue Samples Using Microarray Expression Data. *Bioinformatics* 16, 906–914
5. Antoniadis, A., Lambert-Lacroix, S., Leblanc, F.: Effective Dimension Reduction Methods for Tumor Classification Using Gene Expression Data. *Bioinformatics* 19, 563–570
6. Huang, D.S., Zheng, C.H.: Independent Component Analysis Based Penalized Discriminant Method for Tumor Classification Using Gene Expression Data. *Bioinformatics* 22(15), 1855–1862 (2006)
7. Liao, J.G., Chin, K.V.: Logistic Regression for Disease Classification Using Microarray Data: Model Selection in a Large p and small n case. *Bioinformatics* 23(15), 1945–1951 (2007)

8. Nguyen, D.V., Rocke, D.M.: Tumor Classification by Partial Least Squares Using Microarray Gene Expression Data. *Bioinformatics* 18(1), 39–50 (2002)
9. Belkin, M., Niyogi, P.: Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Comput.* 15(6), 1373–1396 (2003)
10. He, X., Yang, S., Hu, Y., Niyogi, P., Zhang, H.J.: Face Recognition Using Laplacianfaces. *IEEE Trans. Pattern Analysis and Machine Intelligence* 27(3), 328–340 (2005)
11. Zhao, H., Sun, S., Jing, Z., Yang, J.: Local Structure Based Supervised Feature Extraction. *Pattern Recognition* 39, 1546–1550 (2006)
12. Yang, J., Zhang, D., Yang, J.Y., Niu, B.: Globally Maximizing, Locally Minimizing: Unsupervised Discriminant Projection with Application to Face and Palm Biometrics. *IEEE Trans. Pattern Analysis and Machine Intelligence* 29(4), 650–664 (2007)
13. Niyogi, X.H.P.: Locality Preserving Projections. *Advances in Neural Information Processing Systems Vancouver, Canada Univ. of Chicago, Computer Science Department* (2002)
14. Yu, H.C., Bennamoun, M.: 1D-PCA, 2D-PCA to nD-PCA. In: *ICPR 2006, 18th International Conference on Pattern Recognition*, vol. 4, pp. 181–184 (2006)

A Novel Hybrid Method of Gene Selection and Its Application on Tumor Classification

Zhuhong You^{1,2}, Shulin Wang¹, Jie Gui^{1,2}, and Shanwen Zhang¹

¹ Intelligent Computing Lab, Hefei Institute of Intelligent Machines, Chinese Academy of Sciences, P.O. Box 1130, Hefei, Anhui, 230031, China

² Department of Automation, University of Science and Technology of China, Hefei, Anhui, 230027, China

Abstract. Microarray gene expression profile data is used to accurately predict different tumor types, which has great value in providing better treatment and toxicity minimization on the patients. However, it is difficult to classify different tumor types using microarray data because the number of samples is much smaller than the number of genes. It has been proved that a small feature gene subset can improve classification accuracy, so feature gene selection and extraction algorithm is very important in tumor classification. In this paper, a novel hybrid gene selection method is proposed to find a feature gene subset so that the feature genes related to certain cancer can be kept and the redundant genes can be leave out. In the proposed method, we combine the advantages of the PCA and the LDA and proposed a novel feature gene extraction scheme. We also compared several kinds of parametric and non-parametric feature gene selection methods. We use the SVM as the classifier in the experiment and compare the performance of three common SVM kernels. Their differences are analyzed. Using the n-fold cross validation, the proposed algorithm is carried out on three published benchmark tumor datasets and experimental results show that this algorithm leads to better classification performance than other methods.

Keywords: Feature Gene Selection; Tumor Classification; PCA; LDA; SVM; k-NN.

1 Introduction

The development of DNA microarray technology has been produced large amount of gene data and has made it easy to monitor the expression patterns of thousands of genes simultaneously under particular experimental environments and conditions [1]. Microarray gene expression profile data is used to accurately predict different tumor types, which has great value in providing better treatment and toxicity minimization on the patients. Cancer classification using gene expression profiles is becoming popular in both biological and engineering fields in these days. Compared with traditional tumor diagnostic methods mainly based on the morphological appearance of the tumor, the method using gene expression profiles is more accurate and reliable. More

importantly, some tumors, for example lymphoma, have subtypes which have very similar appearances and are very hard to be classified through traditional methods[2][3]. It has been proved that gene expression profile data has good capability to clarify different tumor types and subtypes and this area has become a hot topic because of its promising application.

However, it is difficult to classify different tumor types using microarray data because the number of tissue samples is much smaller than the number of genes. It has been proved that most of the genes may be irrelevant to a specific classification problem and a small feature gene subset can improve classification accuracy, so the important feature genes that contribute most to the reliable classification of cancers need to be identified and selected. Feature genes selection can reduce the number of genes while retaining most of the discriminative information, which will reduce the computational and storage requirements. Finally, feature gene selection also helps to better understand the functions of certain genes. It is very important for designing microarray experiments for clinical diagnosis and prognosis purposes. According above stated, feature genes selection and extraction algorithm is very important in tumor classification and diagnosis[4][5].

The feature selection method can be divided to two categories: filter methods and wrapper methods[3]. In filter methods, genes are selected based on their relevance to certain classes such as statistical tests, Pearson correlation coefficient, etc. The wrapper methods are particular learning algorithms. They can obtain a small subset of genes with high accuracy rate and require extensive computation to search the best features. In addition, the essential difference between these two approaches is that wrapper methods make use of the algorithm that will be used to build the final classifier, while a filter method does not need.

In the past decade, various statistic or machine learning algorithms have been developed for feature gene selection[6][7][8]. In [7], Golub proposed a classical method in which features are ranked according to their linear discriminant ability with an assumption that features are orthogonal to each other. Several kinds of feature gene selection approaches have been proposed in these years and a good review of them can be found in the recent textbook written by Guyon [9].

In this paper, we developed a novel approach to select the relevant feature genes. Different from previous feature gene selection algorithms, the proposed method can obtain the high recognition accuracy rate. Our approach proceeds in three phases. In the first phase we use four kinds of parametric or non-parametric feature gene selection methods to obtain a subset from the gene expression profiles. It provides an initial assessment of a filtering approach. In the second phase, we choose candidate feature gene subsets according to proposed hybrid feature gene extraction algorithm, which combines the advantages of the PCA (Principal Components Analysis) and LDA (Linear Discriminant Analysis). Then the feature subset is delivered to a classifier. This phase substantially reduces the number of feature genes that are input to the SVM classifier. Finally, in the third phase we use Support Vector Machines (SVM) as the classifier. It will be seen that a great accuracy improvement can be achieved by the classification algorithms if discriminatory features are first determined by the above feature selection and extraction methods. We also compare the performance of RBF-SVM classifier with k-NN classifier and their differences are analyzed in the paper.

The remainder of this paper is organized as follows. In Section 2, we present the algorithm description and the classification algorithm is shortly describes in Section 3. Section 4 present the experiments and some results. Finally, Section 5 concludes the paper and gives the discussion and conclusion.

2 Algorithm Description

2.1 Representation of Gene Expression Data

In this paper, the gene expression matrix can be described by $M \times N$ expression matrix G (as defined in Equation 1), in which the horizontal axis represents genes, and the vertical axis represents samples and conditions.

$$G = (x_{ij}) = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1N} \\ x_{21} & x_{22} & \dots & x_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ x_{M1} & x_{M2} & \dots & x_{MN} \end{pmatrix} \tag{1}$$

where x_{ij} denotes the expression level of sample j for gene i , such that $i = 1, 2, \dots, M$ and $j = 1, 2, \dots, N$. For convenience, we consider the tumor classification problem as follows. Suppose there are k classes, labeled as G_1, G_2, \dots, G_k . For tumor classification, we randomly select n_i samples from each class $G_i, i = 1, 2, \dots, k$ to train the classifier, where $G_k : x_1^{(k)}, x_2^{(k)}, \dots, x_{n_k}^{(k)}$.

2.2 Gene Ranking Approach

The problem with existing gene ranking approaches for selecting feature genes is that many of these genes are highly correlated. Feature selection algorithm selects a subset from a complete set of features and can increase the efficiency of the classifier by reducing redundant and irrelevant features[10]. It can be formally defined as follows.

Let S be a subset of X and $S = \{s_1, s_2, \dots, s_n \mid s_i \in X, n \ll \|X\|\}$, where n is the number of features in the subset. The feature selection function F selects s_i from X , that is $F : X \rightarrow S$.

In this paper we adopt four parametric or non-parametric feature gene selection methods: the B-statistic method, the T-statistic method, the Revised Feature Score Criterion (RFSC) and the Wilcoxon Rank-Sum test method. By performing feature selection on the microarray features, it is possible reduce redundancy by excluding features that are not significant. The following sections briefly describe these three statistical methods. The first three are parametric test statistics, while the last one is nonparametric.

2.2.1 B-Statistic

The definition of this test is:

$$B(x) = \frac{\sum_{i=1}^k n_i (u_i - u)^2}{\sum_{i=1}^k (1 - \frac{n_i}{n}) s_i^2} \tag{2}$$

Where n_i is the number of samples from class $G_i, i = 1, 2, \dots, k$. n is the number of all samples. The mean value of the samples from G_i can be calculated by

$$u_i = \sum_{j=1}^{n_i} x_{ij} / n_i . \text{The mean value of all samples can be calculated by } u = \sum_{j=1}^n x_{ij} / n .$$

Also, the standard deviation of the samples from G_i can be expressed by

$$s_i = \left(\frac{1}{n_i - 1} \sum_{j=1}^{n_i} (x_{ij} - u_i)^2 \right)^{\frac{1}{2}} .$$

2.2.2 Revised Feature Score Criterion

Consider a binary classification problem with training instances and its class label pairs (X_j, y_j) , where $j = 1, 2, \dots, N$ and $y_j \in \{-1, 1\}$. The X_j is called a positive instance if the corresponding label is +1; otherwise, it is a negative instance. The Revised Feature Score Criterion is given by:

$$RFSC(g_i) = 0.5 \left| \frac{u_i^+ - u_i^-}{\sigma_i^+ + \sigma_i^-} \right| + 0.5 \ln \left(\frac{(\sigma_i^+)^2 + (\sigma_i^-)^2}{2\sigma_i^+ \sigma_i^-} \right) . \tag{3}$$

Where u_i^+ and u_i^- are the mean values of the samples from positive and negative instance respectively. σ_i^+ and σ_i^- are the standard deviation of the samples from positive and negative instance respectively.

2.2.3 T-Statistic

The definition of this test is:

$$T(g_i) = \frac{u_i^+ - u_i^-}{S_w \sqrt{\frac{1}{n^+} + \frac{1}{n^-}}} \tag{4}$$

Where $S_w = \frac{(n^+ - 1)(\sigma_i^+)^2 + (n^- - 1)(\sigma_i^-)^2}{n^+ + n^- - 2}$, the meaning of $u_i^+, u_i^-, \sigma_i^+, \sigma_i^-$ is the same as above explanation.

2.2.4 Wilcoxon Rank-Sum Test

The Wilcoxon rank-sum test is a nonparametric alternative to the two sample t-test which is based on the order in which the observations from the two samples fall[11]. The Wilcoxon rank-sum test involves comparisons of differences between measurements, so it requires that the data are measured at an interval level of measurement. However, it does not require assumptions about the form of the distribution of the measurements. It should therefore be used whenever the distributional assumptions that underlie the t-test cannot be satisfied.

The general idea of Wilcoxon rank-sum test is that, instead of using the original observed data, we can list the data in the value ascending order, and assign each data item a rank, which is the place of the item in the sorted list. Then, the ranks are used in the analysis. Using the ranks instead of the original observed data makes the rank sum test much less sensitive to noises than the classical parametric tests. A noise will change the t-statistic value greatly, but not much to the ranks. A gene expression data set often has many noises. Thus, it is more suitable to apply the rank sum test on informative gene selection[12].

Given a data set of two classes. The statistic W is the sum of ranks of the samples in the smaller class. The major steps of the Wilcoxon rank sum test are as follows. Firstly, we combine all observations from the two populations and rank them in value ascending order. Secondly, we add all the ranks associated with the observations from the smaller group. This gives the Wilcoxon statistic; Finally, the p-value associated with the Wilcoxon statistic is found from a statistics toolkit, such as Matlab.

2.3 Feature Gene Extraction Algorithm

The following sections briefly describe the methods of Principal component analysis (PCA) and Linear Discriminant Analysis (LDA), which is the theoretical foundation of the proposed algorithm.

2.3.1 PCA Algorithm

PCA is a technique used to reduce multidimensional data sets to lower dimensions for analysis. It is a widely used data analysis technique that allows reducing the dimensionality of the system while preserving information on the variable interactions[13][14]. PCA method transforms the original variables into a set of linear combinations, the principal components (PC), which capture the data variability, are linearly independent and weighted in decreasing order of variance coverage.¹¹ This allows a straightforward reduction of the data dimensionality by discarding the feature elements with low variability. Thus, all original M -dimensional data patterns can be optimally transformed to data patterns in a feature space with lower dimensionality.

The basic approach in PCA is conceptually quite simple. Given matrix $G=(x_{ij})$, where x_{ij} denotes the expression level of sample j for gene i , such that $i = 1, 2, \dots, M$ and $j=1, 2, \dots, N$. Firstly, the M -dimensional mean vector u_i and $M \times M$ covariance matrix Σ are computed for the full data set. Next, the eigenvectors and eigenvalues are computed, and sorted according to decreasing eigenvalue. Call these eigenvectors e_1 with eigenvalue λ_1 , e_2 with eigenvalue λ_2 , and so on. Next, the largest k such

eigenvectors are chosen. In practice, this is done by looking at a spectrum of eigenvectors. The largest eigenvalues correspond to the dimensions that explain larger amounts of variance of the dataset. Form a $M \times k$ matrix A whose columns consist of the k eigenvectors. Then the k -dimensional feature space ($k < M$) can be transformed by: $Y = A^T G(x)$. It has been proved that this representation minimizes a squared error criterion.

2.3.2 LDA Algorithm

Linear Discriminant Analysis (LDA) is another technique that has been successfully used for gene expression profile data analysis. LDA aims to find a set of projection vectors that maximize the between-class scatter and simultaneously minimize the within-class scatter, there by achieving maximum discrimination[15].

Let $X \in R^{p \times n}$ be a sample matrix containing $x_i \in R^p, i = 1, \dots, n$ as columns, where p is the number of features, and n is the number of samples. The between-class scatter matrix S_b and the within-class scatter matrix S_w are defined as:

$$S_b = \frac{1}{n} \sum_{k=1}^c n_k (m^{(k)} - m)(m^{(k)} - m)^T . \tag{5}$$

$$S_w = \frac{1}{n} \sum_{k=1}^c \sum_{j=1}^{n_k} (x_j^{(k)} - m^{(k)})(x_j^{(k)} - m^{(k)})^T . \tag{6}$$

where c is the number of classes, n_k is the number of samples in class k , $x_j^{(k)}$ is the j th sample in class k . $m^{(k)}$ and m are the mean vector of class k and the total mean vector, respectively. Then, classical LDA finds the projection matrix W that maximizes the Fisher criterion:

$$J_{LDA}(W) = \text{trac}((W^T S_w W)^{-1} (W^T S_b W)) \tag{7}$$

subject to $W^T W = I$. By solving a generalized eigenvalue problem, W can be found as the eigenvectors of $S_w^{-1} S_b$ corresponding to the largest eigenvalues. However, when the dimensionality of samples is larger than the sample size, i.e. $p > n$, S_w becomes singular and we cannot compute $S_w^{-1} S_b$, which is a major drawback of classical LDA. This is known as the singularity problem or the small sample size problem.

2.3.3 Our Proposed Hybrid Approach

In this paper, we propose a novel hybrid approach to select a feature gene subset from microarray data so that the feature genes related to particular cancer can be kept and the redundant genes can be leave out. Our approach sequentially combines gene

ranking method and feature extraction approach. In this approach, we first applied gene ranking algorithms to select a set of top-ranked informative genes and then applied proposed hybrid gene selection algorithm which combines the PCA and LDA. Finally, selected genes are classified using SVM and KNN classifier. When applied to two benchmark cancer datasets for tumor classification, our approach is capable of selecting very small sets of informative genes without sacrificing classification accuracy.

The PCA method can transform the original space feature to a new uncorrelated feature set, but it is unable to provide useful class information for classification. On the other hand, the LDA can form the feature set with more discriminability, but it is instable and can not guarantee that features are uncorrelated with each other. Many of feature extraction methods using PCA discard the discriminant information contained in the within-class scatter matrix, but this discriminant information is very effective for the classification problem. PCA method can compute a projection vector which has the largest variance, while LDA method can compute a vector that best discriminates between two class. PCA gives the optimal projection vector that have the largest variance and just select part of eigenvectors corresponding to the value of the eigenvalues. Actually, some PCs that may be benefited to the classification are thrown away if we only apply the PCA method. It is probably that the PCs which contain the information of discriminatory are thrown away after dimension reduction using PCA. On the other hand, the with-class scatter matrix S_w contains discriminative information for classification using LDA. This kind of information is ignored by PCA approach. When the dimension of dataset is not very large and the number of sample is not relatively small, the performance of LDA gains an advantage over PCA. The largest variance direction was not always the best direction for classification (Fig. 1 shows an example).

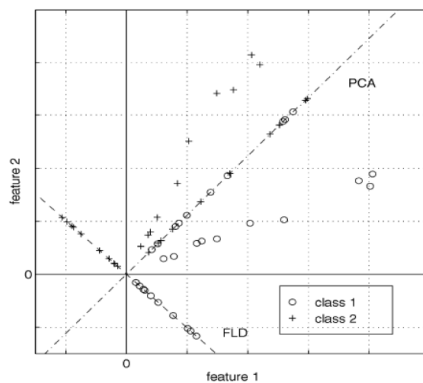


Fig. 1. Shows two class classification instances, marked by circles and crosses. Taking all of the data into account, PCA will compute a vector that has the largest variance associated with it. This is shown by the diagonal line labeled PCA. On the other hand, LDA will compute a vector which best discriminates between the two classes. This vector is shown by the diagonal line labeled LDA.

Combining the methods of PCA and LDA, we develop a novel algorithm, which can give the classifier more valuable discriminative and achieve high classification accuracy.

The rationale of proposed hybrid gene extraction method is to combine the advantages of the PCA and LDA. After eigen-decomposing in PCA, we select those principal components (PCs) with more individual discriminant information instead of those PCs with the highest variance in traditional PCA. By this way, the most valuable information about the discernment between the classes will be obviously added. In particular, our method can guarantee that the PCs are uncorrelated. At the same time, the robust estimate for the PCs based on our method can guarantee that those selected PCs are more efficient for classification.

In summary of the discussion so far, our approach is given as below:

Step 1 (gene ranking): We firstly adopt the three parametric or non-parametric feature gene selection methods one by one: the T-statistic method, the Revised Feature Score Criterion (RFSC) and the Wilcoxon Rank-Sum test method to select a small feature gene subset. Let's take RFSC method for example. For each gene g_i in G , we calculate the mean u_i^+ , u_i^- and standard deviation σ_i^+ , σ_i^- which correspond to the gene g_i of samples labeled +1, -1 respectively. Then we calculate a feature score as equation (xx) for each $g_i \in G$, then rank the genes according to their score values. At last, we simply select the genes with the highest $RFSC(g_i)$ scores as our top genes subset G_{top} , where $|G_{top}| \ll |G|$. In this paper, the number of selected gene subset is about 300.

Step 2 (obtain all PCs using PCA): Applying PCA method to the top ranking genes subset G_{top} , we transform the gene subset to a new uncorrelated feature set in which all PCs corresponding to non-eigenvalues are kept.

Step 3 (select a portion of informative PCs with high discrimination power by introducing LDA method): After Step 2, we have obtained a number of uncorrelated PCs.

Applying LDA criterion $J(V) = \frac{V^T S_B V}{V^T S_w V}$ described above, we can choose those in-

formative PCs with high discrimination power as selected features. Apparently, the higher the discriminant power is, the more powerful the PC is in classification. When the number of the selected features k is specified, we can choose the first k discriminant PCs which have largest discriminant power for classification task.

Step 4 (classification using SVM and KNN classifier): After selecting the feature genes, we classify the test data using one of the following classifiers: k-Nearest Neighbor (k-NN) and Support Vector Machine (SVM). The classification results of these algorithms are then used to evaluate the effectiveness of our proposed hybrid feature gene selection method.

3 Classification Algorithms

We use k-Nearest Neighbor (k-NN) and Support Vector Machine (SVM) classifier in our study. In this section we provide a brief description of these classifiers.

3.1 Support Vector Machine

Support Vector Machine (SVM) introduced by Vapnik has attracted much research attention in these years due its demonstrated improved generalization performance over other techniques in many real world applications including the analysis of microarrays[16][17]. SVM is a relatively new type of statistic learning theory. The SVM originated from the idea of the structural risk minimization developed by Vapnik [13]. SVM is an effective algorithm to find the maximal margin hyperplane to separate two classes of patterns. The main difference between this technique and many other conventional classification techniques including neural networks is that it minimizes the structural risk instead of the empirical risk. The principle is based on the fact that minimizing an upper bound on the generalization error rather than minimizing the training error is expected to perform better. A transform to map nonlinearly the data into a higher dimensional space allows a linear separation of classes, which could not be linearly separated in the original space. The objects that are located on these two hyperplanes are the so-called support vectors. The maximal margin hyperplane, which is uniquely defined by the support vectors, gives the best separation between the classes[18].

SVM realizes the following basic idea: it maps the input vector $x \in X^n$ through kernel function $K(x, y)$ into the high dimension feature space $\Phi(x)$ and then use the maximal margin to build linear machine in the feature space. The final decision function given by an SVM could be expressed as:

$$f(x) = w\Phi(x) + b = \sum_i \alpha_i y_i K(x_i, x) + b. \quad (8)$$

In this paper, radial basis functions (RBF) $K(x_i, x_j) = \exp(-\frac{\|x_i - x_j\|^2}{2\sigma^2})$ kernel is ap-

plied, which has better boundary response and most high-dimensional data sets can be approximated by Gaussian like distributions. In the experiment, we use the well known software LIBSVM to classify the tumor dataset[19].

3.2 k-Nearest Neighbor (k-NN)

The k-NN classifier is a well-known nonparametric classifier. It is based on a simple and effective supervised classification technique[10]. To classify a new input X , the k-NNs are retrieved from the training data. Given an input vector X , k-NN extracts

k closest vectors in the reference set based on similarity measures, and the input X is then labeled with the majority class label corresponding to the k-NN.

Pearson's coefficient correlation and Euclidean distance have been used as the similarity measure. When we have an input X and a reference set $D = \{d_1, d_2, \dots, d_N\}$, the probability that X may belong to class c_j , $P(X, c_j)$ is defined as follows:

$$P(X, c_j) = \sum_{d_i \in kNN} S(X, d_i)P(d_i, c_j) - b_j. \quad (9)$$

where $S(X, d_i)$ is the similarity between X and d_i , and b_j is a bias term.

The k-NN classifier possesses attractive properties including the following: 1) it requires only one parameter—the number of nearest neighbors; 2) it does not need any a priori knowledge about the probability distributions of the classification problem.

4 Experiments and Results

In this section, we present results on two public benchmark tumor datasets. Experiment results are provided in the following sections.

4.1 Tumor Datasets

There are several published benchmark cancer gene expression dataset, including leukemia cancer dataset, colon cancer dataset, lymphoma dataset, breast cancer dataset, and ovarian cancer dataset. In this study, we choose the first two dataset that are briefly described as below to evaluate the performance of proposed algorithm in the experiments and compare the results with others.

1) ALL/AML Leukemia tumor dataset (7129 genes, 72 samples in two classes)

Leukemia dataset consists of 72 samples: 25 samples of acute myeloid leukemia (AML) and 47 samples of acute lymphoblastic leukemia (ALL). The source of the gene expression measurements was taken from 63 bone marrow samples and 9 peripheral blood samples. Gene expression levels in these 72 samples were measured using high density oligonucleotide microarrays. Each sample contains 7129 gene expression levels. This dataset can be downloaded from <http://www-genome.wi.mit.edu/cgi-bin/cancer/publications>

2) Colon cancer dataset (2000 genes, 62 samples in two classes)

Colon dataset consists of 62 samples of colon epithelial cells taken from colon-cancer patients. Each sample contains 2000 gene expression levels. Although original data consists of 6000 gene expression levels, 4000 out of 6000 were removed based on the confidence in the measured expression levels. 40 of 62 samples are colon cancer samples and the remaining are normal samples.

Each sample was taken from tumors and normal healthy parts of the colons of the same patients and measured using high density oligonucleotide arrays. This tumor dataset can be downloaded from <http://microarray.princeton.edu/oncology/affydata/index.html>.

4.2 Results

We use the RBF-SVM and the k-NN as classifiers in the experiments and compare the n-fold cross-validation accuracy for each of the four gene ranking methods (B_statistic, T_statistic, RFSC and Wilcoxon). Firstly, we used each of these methods to select the top-80 ranked genes. Then, we apply PCA method to transform the ranked genes to a new uncorrelated feature set in which all PCs corresponding to non-eigenvalues are kept. After that, we compared the n-fold cross-validation accuracy of the four gene ranking methods. (The k-fold cross-validation relies on a random partitioning of the dataset into k parts. Then, one part is used for testing when the k-1 remainders are used for training.)

Fig. 2 shows the results. We can observe from the results that the performance of Wilcoxon rank-sum test is slightly better than other methods on the Leukemia data set. However, on the colon data set the performance of the four gene ranking methods is comparable. Of the four proposed gene ranking methods, overall speaking, the wilcoxon rank-sum test appeared to be the best.

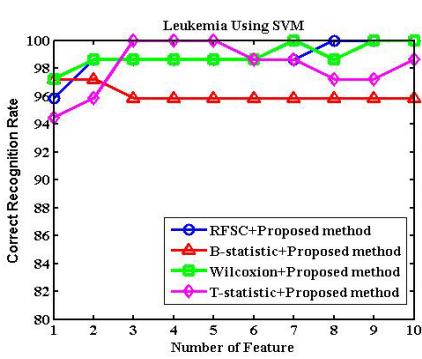
This paper also compared the performance of our proposed method with traditional single gene selection+PCA method on two benchmark tumor datasets using two kinds of classifier: SVM or k-NN. In order to compare the performance of these two approaches, we performed two series experiments.

1) The four gene ranking methods +PCA+SVM (shown in (b), (d) of Fig. 2) or k-NN(shown in (f), (h) of Fig. 2).

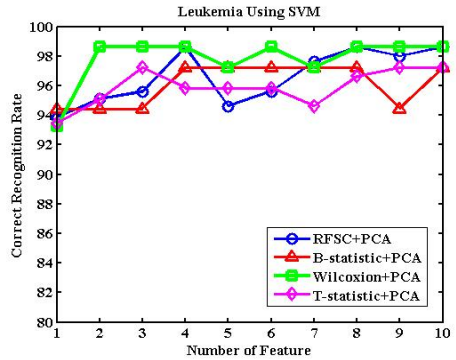
2) The four gene ranking methods +Proposed algorithm+SVM(shown in (a), (c) of Fig. 2) or k-NN(shown in (e), (g) of Fig. 2).

From the classification results of experiments (shown in Fig. 2), we can find that our approach outperforms the traditional method on two data sets. Results show that using the discriminative information of LDA method the proposed method do help in improving the classification accuracy using the same number of feature gene comparing with traditional approach. As we have discussed above, we can conclude that our hybrid method is obviously superior to the single gene selection+PCA method in reducing dimension for SVM classification and in improving performance of SVM classification when retaining high recognition rate. It has demonstrated that the approach is able to reduce the number of genes selected and to increase the classification accuracy rate.

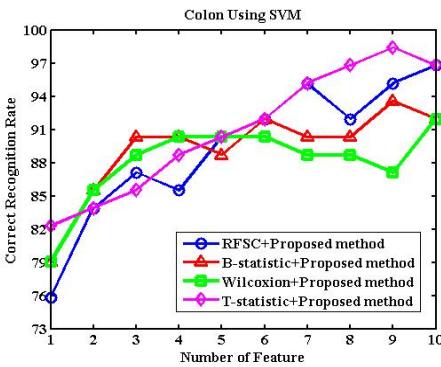
This paper then also compared the performance of SVM classifier (shown in (a), (b), (c), (d) of Fig. 2) with k-NN classifier (shown in (e), (f), (g), (h) of Fig. 2) using the two subsets. Experimental results showed that SVM outperforms the k-NN classifiers. This is why SVM classifier is adopted in this field. Interestingly, the simplest k-NN classifier also provided us a very good accuracy. This indicates that after feature selection the expression data can be well clustered according to distance.



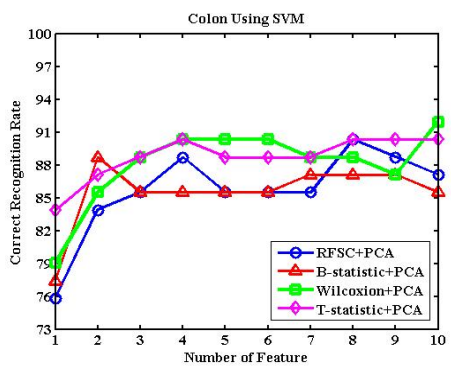
(a)



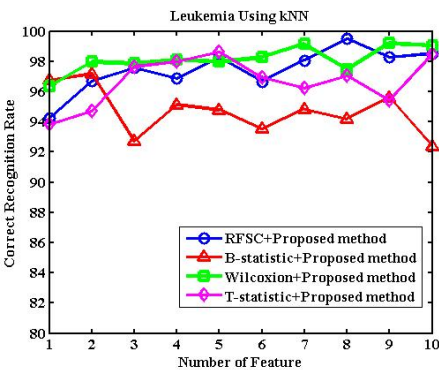
(b)



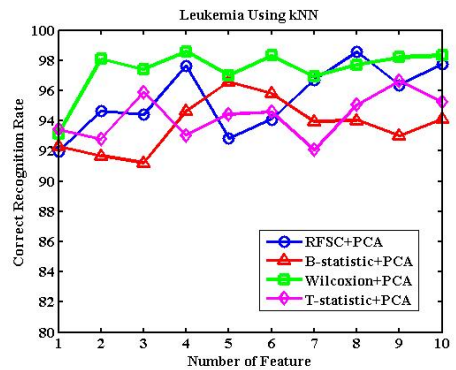
(c)



(d)



(e)



(f)

Fig. 2. The comparison results of the n-fold cross validation accuracy. (a)-(d) Using radial basis functions SVM. (e)-(h) Using k-NN.

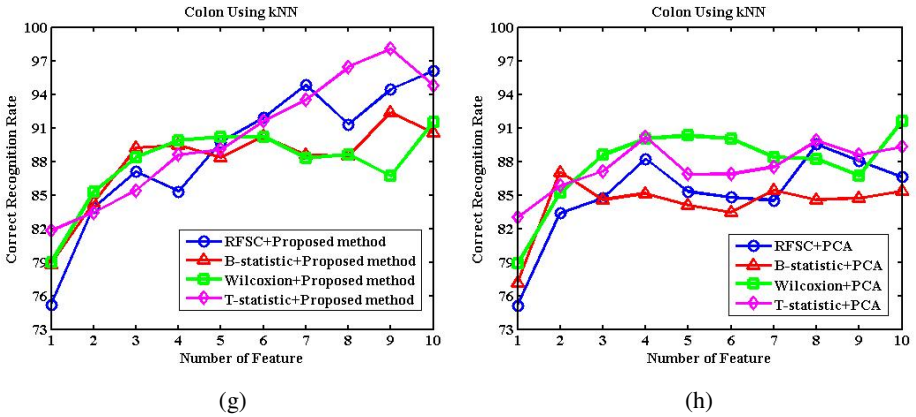


Fig. 2. (continued)

5 Discussion and Conclusion

The main contribution of this paper is to introduce a novel hybrid gene selection algorithm, which sequentially combines four kinds of gene ranking approaches, PCA method and LDA method to select marker genes for classification. The proposed approach has been implemented and evaluated on two public datasets. Experiments showed the good n-fold accuracy using relatively few informative genes (100% on leukemia dataset using 3 genes by the wilcoxon+proposed method+SVM classifier, 97.9% on colon tumor dataset using 9 genes by the wilcoxon+proposed method+SVM classifier. Shown in (a), (c) of Fig. 2). Using far fewer genes, our approach has been shown to be able to offer better n-fold accuracy compared with conventional approaches using all of the top-50 or the top-100 ranked genes directly.

Selecting small number of genes while keeping high classification accuracy is an important goal of concerned problem. Our experimental results illustrate that the proposed method improves the performance on the gene expression microarray data in the accuracy.

This paper also empirically compared the performance of the four gene ranking methods (B_statistic, T_statistic, RFSC and Wilcoxon) for selecting informative genes for cancer classification using microarray gene expression data. Experimental results show that the performance of Wilcoxon rank sum-test is better than other methods in most cases.

Of the two proposed classification algorithms, experimental results show that SVM appeared to perform better than the k-NN classifier. This is why SVM classifier is widely adopted in this field. Interestingly, the simplest k-NN classifier also provided us a very good accuracy. This indicates that after feature selection the expression data can be well clustered according to distance.

Acknowledgements. This work was supported by the grants of the National Science Foundation of China, No. 30700161, the grant of the Guide Project of Innovative Base of Chinese Academy of Sciences (CAS), No.KSCX1-YW-R-30, and the grant of Oversea Outstanding Scholars Fund of CAS, No.2005-1-18.

References

1. Harrington, C.A., Rosenow, C., Retief, J.: Monitoring Gene Expression Using DNA Microarrays. *Int. J. Current Opinion in Microbiology* 3(3), 285–291 (2000)
2. Patra, J.C., Lim, G.P., Meher, P.K.: DNA Microarray Data Analysis: Effective Feature Selection for Accurate Cancer Classification. In: *IJCNN 2007*, pp. 260–265 (2007)
3. Kohavi, R., John, G.H.: Wrapper for Feature Subset Selection. *Artif. Intell.* 97(1/2), 273–324 (1997)
4. Zhang, H.P., Yu, C.Y., Singer, B., Xiong, M.M.: Recursive Partitioning for Tumor Classification with Gene Expression Microarray Data. *PNAS* 98(12), 6730–6735 (2001)
5. Chu, W., Ghahramani, Z., Falciani, F., Wild, D.L.: Biomarker Discovery in Microarray Gene Expression Data with Gaussian Processes. *Bioinformatics* 21(16), 3385–3393 (2005)
6. Brown, M.P.S., Grundy, W.N., Lin, D., Cristianini, N., Sugnet, C., Agnes, J.M., Haussler, D.: Support Vector Machine Classification of Microarray Gene Expression Data. Technical Report, U. California (Santa Cruz) (1999)
7. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S.: Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. *Science* 286, 531–537 (1999)
8. Guyon, I., Weston, J., Barnhill, S.: Gene Selection for Cancer Classification Using Support Vector Machines. *Mach. Learn.* 46, 389–422 (2002)
9. Guyon, I., Elisseeff, A.: An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 1157–1182 (2003)
10. Wang, Y.H., Makedon, F.S., Ford, J.C., Pearlman, J.: HykGene: A Hybrid Approach for Selecting Marker Genes for Phenotype Classification Using Microarray Gene Expression Data. *Bioinformatics* 21(8), 1530–1537 (2005)
11. Deng, L., Pei, J., Ma, J., Lee, D.L.: A Rank Sum Test Method for Informative Gene Discovery. In: *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2004)*, Seattle, WA, USA, pp. 22–25 (2004)
12. Lehmann, E.L.: *Non-parametrics: Statistical Methods Based on Ranks*. Holden-Day, San Francisco (1975)
13. Liu, Z.Q., Chen, D.C., Bensmail, H.: Gene Expression Data Classification with Kernel Principal Component Analysis. *Journal of Biomedicine and Biotechnology*, 155–159 (2005)
14. Jolliffe, I.T.: *Principal Component Analysis*, 2nd edn. Springer, New York (2002)
15. Nijijima, S., Okuno, Y.: Laplacian Linear Discriminant Analysis Approach to Unsupervised Feature Selection. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (to appear, 2008)
16. Vapnik, V.N.: *Statistical Learning Theory*. Wiley, New York (1992)
17. Burges, C.: *A Tutorial on Support Vector Machines for Pattern Recognition*. Kluwer Academic Publishers, Dordrecht (1998)
18. Wang, S.L., Wang, J., Chen, H.W., Tang, W.S.: The Classification of Tumor Using Gene Expression Profile Based on Support Vector Machines and Factor Analysis. In: *Intelligent Systems Design and Applications*, Jinan, China, pp. 471–476. IEEE Computer Society Press, Los Alamitos (2006)
19. Chang, C.C., Lin, C.J.: *LIBSVM: A Library for Support Vector Machines* (2001), <http://www.csie.ntu.edu.tw/~cjlin/libsvm>

An Intelligent Analyzer for Supporting Diagnoses of Congenital CMV Infection

Leonarda Carnimeo

Dipartimento di Elettrotecnica ed Elettronica, Politecnico di Bari,
via E.Orabona, 4-70125 Bari, Italy
carnimeo@deemail.poliba.it

Abstract. Cytomegalovirus (CMV), the most important agent of congenital infection in babies, reveals to be the main cause of not genetic sensorineural hearing loss, that is, the most common congenital kind of hypoacusia. In this paper a neural diagnostic support tool able to identify those children who, from anamnestic/clinic/serologic/virologic point of views, present high risks of developing sensorineural sequelae for CMV infection, is proposed. The adoption of this Intelligent Analyzer is believed to provide a valid contribution to an early diagnosis of this deep form of hypoacusia.

1 Introduction

Hearing problems are the most common children's sensory deficit and sensorineural hearing loss reveals to be the most common congenital kind of hypoacusia. The congenital Cytomegalovirus infection (CMV infection) has been classified as the main not genetic cause of sensorineural hearing loss among all possible causes of pediatric hearing reduction [1]; furthermore, Cytomegalovirus is considered the most important agent of congenital infection in human beings. In reason of the strong correlation between congenital CMV infection and sensorineural hearing loss, the only chance to reduce effects of CMV hypoacusia and of its consequences is constituted by prevention and early diagnosis [2]. Both these factors are considerably influenced by a deep knowledge of the viral natural history and by a set of informations deriving from a strict management of mothers and newborns involved in CMV infection. A large literature on CMV congenital infection has been recently produced to improve the knowledge of this herpetic virus [1-4]; investigation fields and variables are numerous. Moreover, thanks to a multidisciplinary approach to this problem, quite a progress in this diagnostic/clinical field has been achieved. As a consequence, in the last years a significant increase of the amount of medical data has verified, with several difficulties of carrying out a consistent data organization/management adequate for clinical aims.

On this proposal, a great importance is to be given to the knowledge of risk factors [5] and to the availability of fast accurate diagnostic tools which can enable clinicians to identify early risk cases among newborns [6]. From this point of view, Artificial Neural Networks are now considered as effective medical decision support tools [7-9], especially in combined network connections [10].

Starting from the complexity of provided medical informations and the need to correlate anamnestic, virological and clinical data pertinent to mothers in pregravidic and gravidic stage and to their newborns, monitored in intrauterine and neonatal life, in this paper the synthesis of an innovative Intelligent Analyser (IA) is proposed as an effective tool for the management of increasing quantities of not correlated medical data. In detail, the proposed neural diagnostic tool can support CMV infection diagnosis by highlighting some medical data correlations to clinicians and by providing predictive values of sensorineural sequelae risks in infected children, being able to manage increasing amounts of medical data both partial and not complete.

2 State of the Art

In Italy newborns are not subjected to universal hearing screening; thus children deafness deficits are identified nearly at 30 months of life. By behavioural audiometric evaluations, the average value of children's ages identified with hearing loss is reduced to about 13 months, but diagnosis at this age is too late to properly stimulate infant nervous circuits to speech understanding; in fact a delayed intervention beyond 6-12 months of age results in very significant delays in development of communication and language capabilities.

In case of newborns with risk indicators, such as a congenital viral infection, it is concretely possible to advance furtherly on the audiologic diagnosis. However, the selection of fetal disease prognostic markers and of adverse outcomes in newborns remains a major challenge for clinicians and for all those who attend to CMV infected babies' management.

For these reasons, the aim of this work is to develop a suitable Intelligent Analyzer to support a clinical, virological and personal data analysis either for infants or for their mothers. The goal is early to identify newborns with suspected or established CMV congenital infection and provide predictive values of sensorineural sequelae risks linked with viral infection. The choice of some input data rather than others is suggested by some considerations published in recent studies by Fowler, Rivera and Kountakis [1, 3, 5], together with the indications of the Joint Committee on Infant Hearing. Selected anamnestic/serologic/virologic data concern with the more interesting results in the CMV infection assessment both in mothers and in newborns; regarding with maternal determinations, serological tests to define immunological status in pregravidic stage and in pregnancy together with specific prenatal diagnosis results.

Concerning with neonatal determinations, virological issues on neonatal urine samples collected not later than 2 weeks of life are considered, together with proper serum and PMNL sample collection data. Thus, the proposed Intelligent Analyzer, together with the virologic results given by DBS test on neonatal Guthrie Cards [6], reveals able to provide useful risk factor informations for an early identification of congenitally infected newborns.

Concerning with newborns' parameters, considered anamnestic data are Gestational age, Birth weight, Conventional growth parameter and Clinical signs attributable to congenital CMV infection used to classify each examined newborn as Symptomatic or not. It is well known that not all congenitally infected children will develop Sensorineural Hearing Loss (SNHL). This implies that clinical data

regarding prenatal infection alone cannot determine SNHL higher risk babies. Therefore in this work, possible relations between clinical/virological data and SNHL have been investigated as to define which parameters can provide a prediction level of High/Low risk to develop sequelae.

For this purpose, an encrypted database kindly provided by the Laboratory of Virology-Hygiene, Epidemiology and Public Health II Unit - Policlinico di Bari – Italy on the basis of the demographic and virologic findings performed in the last few years in Region Apulia and neighbouring Italian Regions, has been considered [6].

In detail, the complexity of links among provided anamnestic data and virologic direct /indirect results, collected to mothers before and during pregnancy and to their conception products, has been accurately analyzed together with clinicians, always focusing objectives of classification and prediction.

In particular, medical data of 90 children, screened at the Laboratory of Virology - Hygiene, Epidemiology and Public Health at the Policlinico of Bari during a study on congenital CMV Infection/Sensorineural Hearing Loss in Apulia Region in the period 2001-2004 with presumed CMV infection, have been specifically taken into consideration and their well known medical data and pathologies have been isolated. A corresponding input database has been suitably codified in a preprocessing phase before being used for the synthesis of the proposed Intelligent Analyzer.

3 Intelligent Analyzer Architecture

It has to be precised that the Intelligent Analyzer presented in this paper does not aim at substituting audiologic diagnostic methods at all, because only these methods allow a sure identification of deafness. The objective of this system is to identify the cases with a highest risk factor from a clinic/virological point of view.

In other words, due to the complexity of available medical informations and the need to correlate the above mentioned data, the presented contribution defines and develops a novel neural system with a complex neural architecture similar to a biologically hierarchical structure in order to early identify newborns which present a high risk to develop sensorineural sequelae from anamnestic, clinical, serological and virological point of views. Therefore, this neural system can support early diagnosis of deafness by four tasks, each performed by a different block:

- Analysis of clinical symptoms at birth.
- Classification of maternal serologic status.
- Determination of CMV infected newborns.
- Identification of children with high sensorineural sequelae risk.

Due to the independence between some classes of data, a modular organization has been adopted while synthesizing this system, as shown in the behavioural model of the Analyzer reported in Fig.1. In the first block a subsystem, made of information code/decode preprocessing units is present. In the last three blocks, a subsystem, made of information code/decode preprocessing units and a dedicated MLP subnet NN_i , is present and performs one of the four cited tasks.

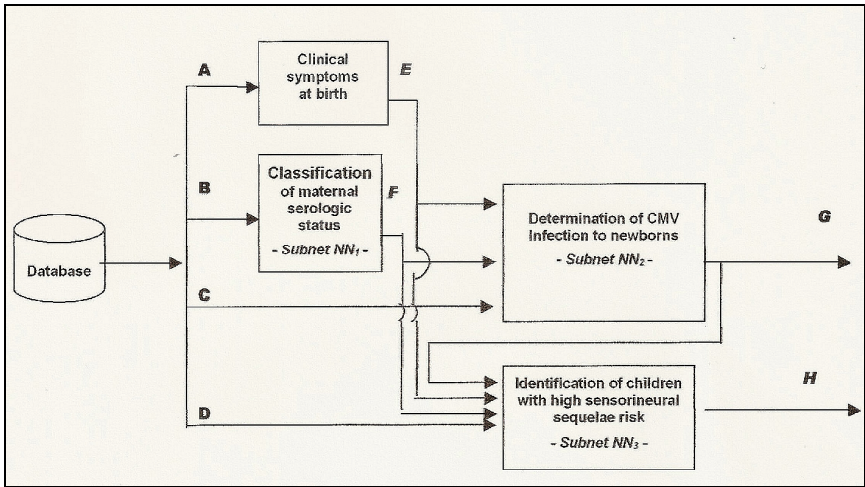


Fig. 1. Block diagram of the proposed Intelligent Analyzer Architecture

Each block considers different data in the whole available database and is properly connected to other blocks. Significant input data have been collected in proper groups by distinguishing mothers' data from newborns' ones from a medical point of view. In detail, four sets of quantities have been considered, which can have a known/unknown value in a database for every clinic case

Input Set A

- 1 - Microcephaly
- 2 - Intracranial calcifications
- 3 - ALT (>80 IU/mL)
- 4 - Chorioretinitis
- 5 - Neurologic abnormalities
- 6 - Petechiae
- 7 - Splenomegaly
- 8 - Hepatomegaly
- 9 - Jaundice
- 10 - Increased CSF protein level
- 11 - Haemolytic Anaemia
- 12 - Thrombocytopenia
- 13 - Hyperbilirubinaemia
- 14 - Small for Gestational Age-SGA

Input Set C

- 1 - DBS test
- 2 - Urine
- 3 - PMNL
- 4 - Serum
- 5 - Newborn's age in days

Input Set B

- 1 - Preconcepnal anti-CMV IgM
- 2 - Preconcepnal anti-CMV IgG
- 3 - Anti-CMV IgM during pregnancy
- 4 - Anti-CMV IgG during pregnancy

Input Set D

- 1 - Gestational period
- 2 - Birth weight
- 3 - Length
- 4 - Cranial Circumference
- 5 - Mother's Age

A double diagnostic tool can be identified both in supporting a correct detection of congenitally infected newborns and in identifying the most exposed ones to the risk of developing sensorineural sequelae. These aids are properly codified by the output values provided by the network. Output E and Output F from the first subsystem and

from the second one are input data for the third block and for the fourth one. Output **G**, which classifies the kind of CMV infection for the generic baby, is also an input for the last block with Output **H**, which is specifically determined to identify cases with high risk of Sensorineural Hearing Loss (SNHL) in congenitally CMV infected babies. This subsystem provides a risk level prediction of SNHL for each child by considering previous outputs **E**, **F**, **G**, like its inputs, together with **D**.

An accurate knowledge of clinical influences among data enables an a priori selection of input data for each subnet of three MLP neural subnetworks NN_1 , NN_2 , NN_3 . Each net, properly trained with a specific algorithm [11], contributes to provide a specific output as follows:

Output E – Clinical signs – Identification of symptomatic/asymptomatic newborns by clinical analyses at birth

Output F - Classification of maternal serologic status - Primary/secondary infection.

All mothers are divided into 6 classes :

A - CMV Sieronegative	D -Presumed Seroconverted
B - Immune	E -Seroconverted
C -Recurrent infected	F -Undefined

Output G - Classification of CMV infection - Types of CMV infection in newborns

- A - Congenital Infection
- B - Postnatal Infection
- C - No Infection
- D - Uncertain Result – Lack of basic analyses/conflicting data - uncertain virologic result

Output H - Risk of SNHL - Identification of congenitally infected children at High Risk

- I - **No Risk** Newborn without congenital infection
- II - **Low Risk** Congenitally infected newborn with low possibilities to develop SNHL
- III - **Attention** Not estimated Risk Level due to lack of medical data
- IV - **High Risk** Congenitally infected newborn with high risk of developing SNHL

Subnet NN_1 - Classification of maternal serologic status

The subnet NN_1 aims at classifying maternal serologic status and has an MLP architecture with 4 input neurons, 3 hidden neurons and 1 output neuron. A transfer Tanh function and the Resilient Propagation learning algorithm [12] are adopted.

Subnet NN_2 - Determination of CMV infection in newborns

This task is performed by synthesizing an MLP subnet NN_2 with 8 input neurons, 6 hidden neurons and 1 output neural cell. A supervised learning procedure has been considered with a training set of 68 patterns and a test set of 30 patterns. Both the hidden layer and the output one are characterized by a transfer function Tanh and a Conjugate Gradient learning algorithm with momentum where $\eta = 0.1$ and $\rho = 0.7$. The supervised training phase involved the maximum value of 10000 epochs and a stop criterion based on a MSE with threshold $T_h = 0.01$ for an on-line weight updating procedure.

The output of this net can assume three values (-1, 0, 1) corresponding to (Absence of infection; Congenital infection; Post-natal infection).

Subnet NN₃ - Identification of children with high SNHL risk

This task is performed by synthesizing an MLP subnet NN₃ with 11 input neurons, 5 hidden neurons and 1 output neural cell. An optimization study has been performed to select the best learning algorithm and determine its parameters. A Gradient Descent Learning Algorithm with momentum revealed the most effective one. It is well known that this solution strongly depends on established parameters. For this reason, the parameter combination which optimize NN₃ performances was evaluated. The output value 0/1 indicates Low/High Risk of SNHL.

4 Results and Discussion

The risk of SNHL for 50 children born in 2005 in the Apulia region and screened for presumed CMV infection has been investigated by taking into account their partially known medical data by means of the synthesized Intelligent Analyzer.

Even if the available input data **A, B, C, D** both from mothers and newborns reveal strongly not complete, nor homogeneous with regard to anamnestic/ clinical/ serological/audiological point of views, the proposed Intelligent Analyzer reveals able to provide a support to medical diagnosis. All available input data of the 4 congenitally infected children are reported in the Table 1.

Among these 50 newborns, the Intelligent Analyzer detects 4 congenital infected children, that is a value of 8%, 23 postnatal infected children (46%), 22 uninfected ones (44%) and 1 with an uncertain result (that is 2%).

Table 1. a

Case ID	Input A		Input B				Input C				
	Symptoms	IgM	IgG	IgM	IgG	WG	DBS	t. Urine	PMNL	Serum	Days
10	2,3,6,7,8 11,12,14	NEG	POS	NEG	POS	13	POS	POS	Un	POS	2
11	0	Un	Un	POS	POS	8	POS	POS	POS	POS	4
30	0	Un	Un	POS	POS	25	Un	POS	POS	POS	4
39	0	Un	Un	POS	POS	34	POS	POS	POS	NEG	60

Table 1. b

Case ID	Input D				
	Gest.period	Birth Weight [g]	Length [cm]	Cranial Circ [cm]	Mother's Age
10	38	2320	45	32	33
11	37	2570	43.	32	27
30	38	3880	48	34	32
39	37	2990	46	34	33

In particular, the IA reveals able to:

- i) detect Case ID 11 as an SNHL **High Risk Case** in absence of symptoms at birth;
- ii) provide a predictive diagnostic support of sensorineural deficit development in ID 30 and ID 39, even in situations of lack or not adequate diagnostic data or loss of clinical symptoms.

These output results are reported in the following Table

Table 2.

Case	Output E	Output F	Output G	Output H
10	Symptomatic	Recurrent Infected	Congenital Infection	Attention
11	Asymptomatic	Presumed Seroconverted	Congenital Infection.	High Risk
30	Asymptomatic	Presumed Seroconverted	Congenital Infection	Attention
39	Asymptomatic	Presumed Seroconverted	Congenital Infection	Low Risk

5 Conclusions

In this work a new neural tool to support virologic diagnosis, called *Intelligent Analyzer* has been presented with two main tasks: a correct detection of congenitally CMV infected newborns and, among these babies, an exact indication of the most exposed ones to the risk of developing sensorineural sequelae. Moreover, on the basis of clinical/virological real data, the proposed Intelligent Analyzer can support an early diagnosis by identifying which infected children will present a higher risk of developing sensorineural hearing loss. Performances of the proposed NN-based system prove satisfactory and indicate the possibility of further improvements as a diagnostic support tool.

Acknowledgments. The author is sincerely grateful to Dr. Agata Calvario of the Laboratory of Virology – Hygiene, Epidemiology and Public Health, Policlinico di Bari for her significant suggestions to this work.

References

1. Fowler, K.B., Dahle, A.J., Boppana, S.B., Pass, R.F.: Newborn Hearing Screening: Will Children with Hearing Loss Caused by Congenital Cytomegalovirus Infection be Missed? *Journ. of Pediatrics* 135(1), 60–64 (1999)
2. Adler, S.P., Finney, J.W., Manganiello, A.M., Best, A.M.: Prevention of Child-to-mother Transmission of Cytomegalovirus among Pregnant Women. *Journal of Pediatrics* 145(4), 485–491 (2004)
3. Rivera, L.B., Boppana, S.B., Fowler, K.B., Britt, W.J., Stagno, S., Pass, R.F.: Predictors of Hearing Loss in Children with Symptomatic Congenital Cytomegalovirus Infection. *Journal of Pediatrics* 110, 762–767 (2002)

4. Calvario, A., Bozzi, A., Scarasciulli, M.L., et al.: Herpes Consensus PCR test: A Useful Diagnostic Approach to the Screening of Viral Diseases of the Central Nervous System. *Journal of Clin. Virology* 25, S71–S78 (2002)
5. Kountakis, S.E., Skoulas, I., Phillips, D., Chang, C.J.: Risk Factors for Hearing Loss in Neonates: A Prospective Study. *Amer. J. of Otolaryngology* 23(3), 133–137 (2002)
6. Calvario, A., Scarasciulli, M.L., Bozzi, A., Ventola, C., Satalino, M., Germinario, C.: Analysis of HCMV Congenital Newborns –DBS Test Positive in Apulia Region. In: 5th National Congress of the Italian Society of Virology (SIV), Orvieto, p. 63 (September 2005)
7. Sargent, D.J.: Comparison of Artificial Neural Networks with Other Statistical Approaches. *Cancer* 91(S8), 1636–1642 (2001)
8. Walczak, S.: Artificial Neural Network Medical Decision Support Tool: Predicting Transfusion Requirements of ER Patients. *IEEE Trans. on Information Technology in Biomedicine* 9(3), 468–474 (2005)
9. Calvario, A., Carnimeo, L., Bozzi, C., Birtolo, M.L.: Scarasciulli Congenital Cytomegalovirus Infection: Risk Analysis via A Neural Analyzer. In: *Int. Conference on Congenital CMV Infection*, Orvieto, Italy (2006)
10. Hayashi, Y., Setiono, R.: Combining Neural Network Predictions for Medical Diagnosis. *Computers in Biology and Medicine* 32(4), 237–246 (2002)
11. Haykin, S.: *Neural Networks: A Comprehensive Foundation*. Prentice Hall, Englewood Cliffs (1999)
12. Riedmiller, M., Braun, H.: A Direct Adaptive Method for Faster Backpropagation Learning, the RPROP Algorithm. In: *IEEE Int. Conf. on Neural Networks*, pp. 586–591 (March 1993)

Biomedical Text Mining Using a Grid Computing Approach

Marcello Castellano^{1,2}, Giuseppe Mastronardi^{1,2}, Giacinto Decataldo¹, Luca Pisciotta¹, Gianfranco Tarricone¹, Lucia Cariello^{1,2}, and Vitoantonio Bevilacqua^{1,2}

¹ Dipartimento di Elettrotecnica ed Elettronica Politecnico di Barivia Orabona,
4 70125- Bari-Italy
castellano@poliba.it

² e.B.I.S. s.r.l. (electronic Business in Security), Spin-Off of Polytechnic of Bari,
Str. Prov. per Casamassima Km. 3-70010 Valenzano (BA)-Italy

Abstract. Extracting useful information from a very large amount of biomedical texts is an important and difficult activity in biomedicine field. Data to be examined are generally unstructured and the available computational resources do not still provide adequate mechanisms for retrieving and analyse very large amount of contents. In this paper we present a rule-based system for Text Mining process applied in biomedical textual documents. This application requires a strongly use of the computational resource to perform intensive operations. We propose a grid computing approach to improve application performance.

Keywords: Text Mining; Computational Grid; SIMD; Knowledge Discovery; Biomedical Document Analysis.

1 Introduction

The problem of discovering useful knowledge from unstructured text, is attracting increasing attention. The process of extracting interesting and not-retrieval patterns or knowledge from unstructured text documents is known as Knowledge Discovery in Text (KDT). One of the most difficult applications of Knowledge Discovery in Texts is Text Mining (TM) of biomedical papers: the sheer volume of biomedical research output makes TM a necessity, while the importance of this research requires extremely high retrieval precision. TM examines the relationships between specific kinds of information contained both within and between documents. TM concentrates on solving a specific problem in a specific domain identified a priori. TM can aid database curators by selecting articles most likely to contain information of interest, or potential new treatments. The goal of biomedical text mining is therefore to allow researchers to identify needed information more efficiently, uncover relationships obscured by the sheer volume of available information, and in general shift the burden of information overload from the researcher to the computer by applying algorithmic, statistical and data management methods to the vast amount of biomedical knowledge [1,2,3,4].

In this paper we discuss a Text Mining Process applied in biomedical documents analysis to recognize biological entities based on rule system. Moreover we discuss a grid computing approach to the problem in order to improve the TM application performance. Finally, experimental results and conclusions are presented.

2 Text Mining for Biomedical Document Analysis

Significant progress has been made in applying text mining to the following text analysis: named entity recognition, text classification, terminology extraction, relationship extraction and hypothesis generation.

The task of named entity recognition appears straightforward. The goal is to identify, within a collection of text, all of the instances of a name for a specific type of thing: for example, all of the drug names within a collection of journal articles, or all of the gene names and symbols within a collection of abstracts. The idea is that recognising biological entities in text allows for further extraction of relationships and other information by identifying the key concepts of interest and allowing those concepts to be represented in some consistent, normalised form. This task has been challenging for several reasons. First, there does not exist a complete dictionary for most types of biological named entities, so simple text matching algorithms do not suffice. In addition, the same word or phrase can refer to a different thing depending upon context. Biological entities may also have multi-word names, like carotid artery, so the problem is additionally complicated by the need to determine name boundaries and resolve overlap of candidate names.

Text classification attempts to automatically determine whether a document or part of a document has particular characteristics of interest, usually based on whether the document discusses a given topic or contains a certain type of information. Typically the information of interest is not specified explicitly by the users and, instead, they provide a set of documents that have been found to contain the characteristics of interest (the positive training set), and another set that does not (the negative training set). Text classification systems must automatically extract the features that help determine positives from negatives and apply those features to candidate documents using some kind of decision-making process.

Paralleling the growth of the increase in biomedical literature is the growth in biomedical terminology. Because many biomedical entities have multiple names and abbreviations, it would be advantageous to have an automated means to collect these synonyms and abbreviations to aid users doing literature searches. Furthermore, other text-mining tasks could be done more efficiently if all of the synonyms and abbreviations for an entity could be mapped to a single term representing the concept. Most of the work in this type of extraction has focused on uncovering gene name synonyms and biomedical term abbreviations.

The goal of relationship extraction is to detect occurrences of a pre-specified type of relationship between a pair of entities of given types. While the type of the entities is usually very specific, like genes, proteins or drugs, the type of relationship may be very general, like any biochemical association, or very specific, like a regulatory relationship.

While relationship extraction focuses on the extraction of relationships between entities explicitly found in the text, hypothesis generation attempts to uncover relationships that are not present in the text but instead are inferred by the presence of other more explicit relationships. The goal is to uncover previously unrecognised relationships worthy of further investigation [5,6].

Fig. 1 shows Knowledge Discovery in Text process steps. Knowledge Discovery in Text can be visualized as consisting of two phases: Text Refining that transforms free-form text document in a chosen Intermediate Form and Text Mining that deduces patterns or knowledge from the Intermediate Form [2,7,8,9,10].

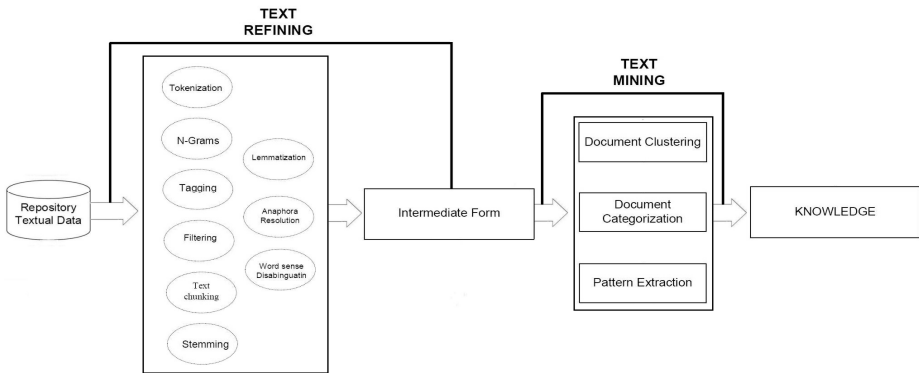


Fig. 1. Knowledge Discovery in Text

2.1 Building the Association Rule Induction

Association discovery is a central activity of the text mining phase. It is the identification process of meaningful correlations among frequent whole data and can be applied to textual structured form. A rule consists of a left-hand side proposition, antecedent, and a right-hand side, consequent. Both sides consist of Boolean statements or predicates. The rule states that if the left-hand side is true, then the right-hand side is also true. A probabilistic rule modifies this definition so that the right-hand side is true with probability p , given that the left-hand side is true. An association rule is described by the form:

$$X \Rightarrow Y$$

where X and Y are predicates or set of items. As the number of produced associations might be huge, and not all the discovered associations are meaningful, two probability measures, called support and confidence, are introduced to discard the less frequent associations in the database.

The support is defined as:

$$fr(X \Rightarrow Y) \quad (1)$$

while the confidence is:

$$c(X \Rightarrow Y) = \frac{fr(X \wedge Y)}{fr(X)} \quad (2)$$

The accuracy of the association rule, when regarded in terms of conditional probability, can be seen as a maximum likelihood (frequency-based) estimate of the conditional probability that Y is true, given that X is true.

Through association rules it happens the classification or text categorization of the analyzed text. The definition of accurate rules allows to the TM tool to analyze the whole text and to decide if the tokens, in which the text is divided, belongs to a class, lemma, or another. Any TM rule founds its principle on the matching, that is to say, "keywords lists" are created each of which contains a subset of words, specifications and not, of the universe, in our case of the Biomedicine. The matching is not enough when one of these words, token, is present in more lists or there isn't in some lists, in such case it needs to understand what is the correct meant to give to the token and therefore the TM rules must be define so that to analyze the tokens that follow or precede the token to classify. The matching rule is the rule that seeks the presence in a keywords lists for every considered token and associates, to the token, the name of the same list:

*IF token belongs to a list AND token it doesn't belong to other lists
THEN the meaning to the token is the name of the same list*

The problem is when the token is present in more or no lists, in such case grammatical rules and interpretation rules are necessary. For example we consider the word "Fanconi" this can be in two different contexts, in the sentence "...Nicola Fanconi restaurant...." or "...discovery of new drugs for the anaemia of Fanconi....", to this point the simple matching rule is not enough to give an unequivocal meaning to the token "Fanconi" being in two lists "Surname" and "Pathology". To give a meaning to this token is needed categorization the lemmas, that is, understanding the meaning of the following and the precedent token, an example is the following rule:

*IF token belongs to the list "surname" AND token belongs to the list "pathology"
IF preceding token OR following token belongs to the list "names"
THEN token preceding/following & token means "first name"
ELSE token means "pathology"*

An example of rule is in which a token is not in any list, in such case it is need to understand what meant it can have, for example, if we consider "Anatomy and Orthopaedics Anderson-Fabry though CT" the token "Anderson-Fabry" could not be present in any list. To this point it is possible to understand which meant to give to this token, in our case, being followed by a key CT (compression therapy), the token is the name of one syndrome that brings the name of the physicians that have discovered her; then a rule of such case is:

*IF token is not present in any list
IF preceding token belongs to the list "therapy/drugs"
THEN token & token preceding/following mean "pathology"*

Another rule is:

*IF token belongs to the list 'human anatomy' AND preceding token belongs to the list "therapy/drugs"
THEN token & token preceding/following mean "symptom"*

3 Grid Computing in Text Mining

In Figure 2a is shown a schema of the typical Job produced by a TM tool. This is a single-instruction and multiple-data stream Job that reveals an intensive use of the CPU resource. In Figure 2b is shown the Job model here adopted to overcome the single CPU bound. It distributes several computing nodes the user commands and a slice of the whole data set. At the completion time each node came back the results.

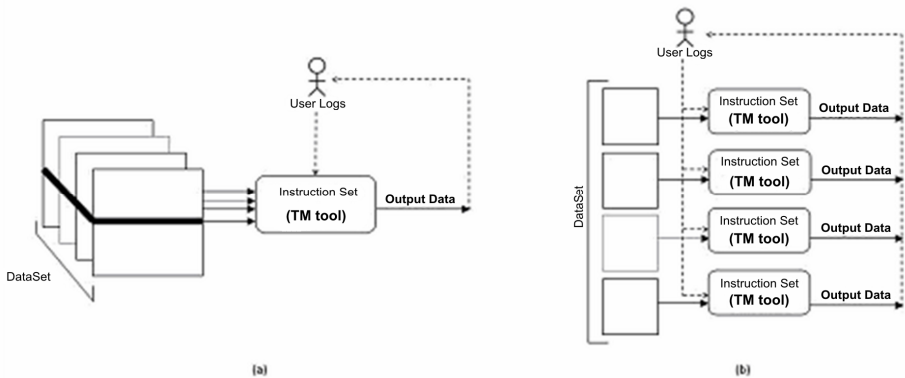


Fig. 2. Text Mining Job Models

In Figure 3 is shown the layered adapter system architecture designed as a software application between a grid middleware infrastructure and the user application to implement the Job model for Grid Computing.

In the hierarchical architecture, there is, to lowest level, the grid middleware with its services, and to next level the shell scripts that interoperate to grid middleware using services.

The functional management system layer is composed by:

- Grid node search system;
- A Load Balancer;
- User program modules ;
- User's module management system;
- A Transfer Optimizer;

The *node search system* effects the search of nodes which are available for execution.

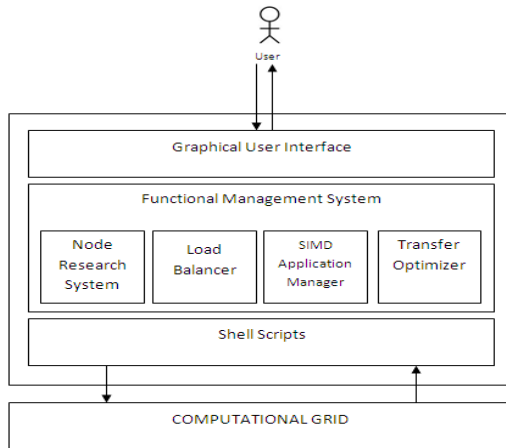


Fig. 3. The Adapter Hierarchical Architecture

The *Load Balancing system* analyzes the whole input data set and the computing nodes previously selected to split the work for parallel execution.

The *user program modules* are developed or available for the end user. They are programs which are distributed on the nodes together data for the execution. It receives and produce a file or a set text file.

The *program modules manager* allows to the user to add and to manage all modules that user wants to use. The manager produce its activity up on a set up of a configuration file. For example to taken into account the applicative modules for the symptoms and pathologies extraction a such configuration file must be updated.

The *transfer optimizer module* performs the data compression optimizing the communication time.

Upper layer, there is a graphical user interface, that supplies the direct and intuitive access to the system functionalities. Through simple click and, in completely transparent way to the user, the interface dialogues with the underlying modules. The communication with lower layer modules, is realized by a shell script passing the actual value of the parameter.

The software system architecture here presented is an effective solution to improve the performance of TM application in bioinformatics.

4 Experimental Results

The process of Text Mining starts from a set of 1000 scientific full text publications available on MedLine / Pubmed in pdf format. The process consists of three phases: text extraction from pdf documents, text mining rules application as described in section 2.1, results storage in a repository composing of identified biological entities (symptoms and pathologies). Text mining rules application is realized using GATE 4.0 tool JAVA APIs [13]. Figure 4 shows a sample of obtained text mining results. Results show the efficiency of the rules application.

Grid execution of text mining operation has been realized through a prototype development written in Java following the system architecture proposed in paragraph 3, using Globus Toolkit 4 grid middleware [14].

The use of standard Java APIs allows porting code on all computational grid nodes used for this experiment, provided that the experimental grid is here built using Globus Toolkit 4 on Gnu / Linux systems.

System efficiency has been evaluated through execution time estimated on a serial computer and then on a computational grid. Execution time has been estimated for a grid with a variable nodes number considering an input data set composed of 1000 biomedical documents with appropriate dimensions. Estimation of speedup indicator has been obtained. Speedup indicator is defined as the ratio between the mining process execution time on traditional serial machine, and on a computational grid. Figure 5 shows architecture speedup depending on grid node number.

Biomedical Document	Identified Pathology
<i>The prevalence of tuberculosis in the state of acre</i>	tuberculosis
<i>The prevalence of tuberculosis in the state of acre</i>	death
<i>The prevalence of tuberculosis in the state of acre</i>	aids
<i>Potentially infectious residues at hemotherapy services [...]</i>	hepatitis
<i>The meaning of cancer in the everyday of women [...]</i>	cancer
Biomedical Document	Identified Symptom
<i>Haemophilus influenzae antibiotic resistant strains [...]</i>	sinusitis
<i>Haemophilus influenzae antibiotic resistant strains [...]</i>	bronchitis
<i>Haemophilus influenzae antibiotic resistant strains [...]</i>	pneumonia
<i>Efficacy of cow's kamiss in the treatment of large [...]</i>	regression
<i>Association between particulate air pollution and first [...]</i>	regression

Fig. 4. Text Mining Results

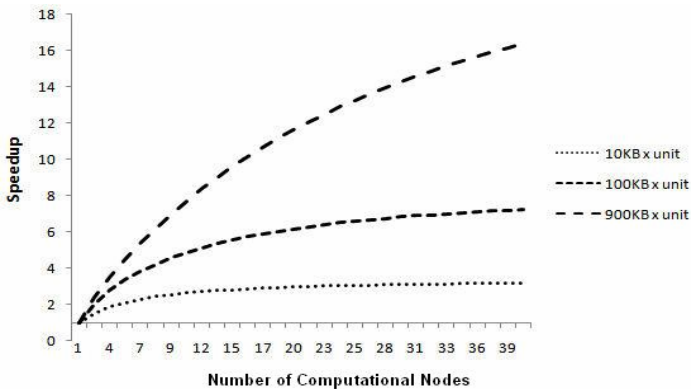


Fig. 5. Speedup depending on grid node number, considering a data set of 1000 documents from 10KB to 900KB each

5 Conclusions

In this paper has been presented a biomedical Text Mining using a grid computing approach. The TM application requires a strongly use of the computational resource and we proposed a grid based approach to improve the application performance.

References

1. Polajnar, T.: Survey of Text Mining of Biomedical Corpora (2006)
2. Tan, A.H.: Text Mining: The State of the Art and the Challenges. In: Zhong, N., Zhou, L. (eds.) PAKDD 1999. LNCS (LNAI), vol. 1574. Springer, Heidelberg (1999)
3. Prather, J.C., Lobach, D.F., Goodwin, L.C., Hales, J.W.: Medical data mining: knowledge discovery in a clinical data warehouse. In: Proc. AMIA Annu. Fall Symp, Division of Medical Informatics, Duke University Medical Center, Durham, North Carolina, USA, pp. 101–105 (1997)
4. Cohen, A.M., Hersh, W.R.: A Survey of Current Work in Biomedical Text Mining. Briefing in Bioinformatics 6 (2005)
5. Polanski, A., Kimmel, M.: Bioinformatics. Springer, Heidelberg (2007)
6. Hersh, W.: Evaluation of biomedical text-mining systems. Briefings in Bioinformatics 6(4), 344–356 (2005)
7. Manning, C.D., Schütze, H.: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge (1999)
8. Ahonen, H.: Finding All Maximal Frequent Sequences in Text. In: ICML 1999 Workshop on Machine Learning in Text Data Analysis, Bled, Slovenia (1999)
9. Hotho, A., Numberger, A., Paab, G.: A Brief Survey of Text Mining. LDV Forum-GLDV Journal for Computational Linguistics and Language Technology 20(1), 19–62 (2005)
10. Mobasher, B., Cooley, R., Srivastava, J.: Creating Adaptive Web Sites Through Usage-Based Clustering of URLs (1999). In: Proc. of the 1999 IEEE Knowledge and Data Engineering Exchange Workshop (KDEX 1999) (1999)
11. Foster, I., Kesselmann, C.: The Grid: Blueprint for a New Computing Infrastructure. Morgan-Kaufmann edition (1998)
12. Castellano, M., Aprile, A., Mastronardi, G., Piscitelli, G., Dicensi, V., Giuseppe, D.G.: Simulating a Computational Grid. GESTS, International Transaction on Communication and Signal Processing (2007)
13. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In: Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL 2002), Philadelphia (2002)
14. The Globus Alliance: Globus Toolkit 4, <http://www.globus.org/toolkit>

High-Throughput Analysis of the Drug Mode of Action of PB28, MC18 and MC70, Three Cyclohexylpiperazine Derivative New Molecules

Vitoantonio Bevilacqua¹, Paolo Pannarale¹, Giuseppe Mastronardi¹, Amalia Azzariti², Stefania Tommasi², Filippo Menolascina¹, Francesco Iorio^{3,5}, Diego Di Bernardo³, Angelo Paradiso², Nicola A. Colabufo⁴, Francesco Berardi⁴, Roberto Perrone⁴, and Roberto Tagliaferri⁵

¹ Dipartimento di Elettrotecnica ed Elettronica, Polytechnic of Bari,
via E. Orabona, 4,70125, Bari, Italy
bevilacqua@deemail.poliba.it

² IRCCS Ospedale Oncologico di Bari
Via Samuel F.Hahnemann, 10 70126 Bari, Italy
a.azzariti@oncologico.bari.it

³ TIGEM - TeleThon Institute of Genetics and Medicine
Via Pietro Castellino, 111 80131 – Napoly, Italy
iorio@tigem.it

⁴ Department Farmaco Chimico, University of Bari,
via E. Orabona, 4,70125, Bari, Italy
colabufo@farmchim.uniba.it

⁵ Department of Mathematics and Informatics, University of Salerno,
Via Ponte Don Melillo, 84084 Fisciano (SA), Italy
rtagliaferri@unisa.it

Abstract. Objective: This work explores the mode of action of PB28, MC70 and MC18 three molecules that showed anti-tumoral properties by arresting cellular growth and inhibiting glycoprotein P. Methods: Here we conduct a microarray-based study and analyze the expression patterns associated with the action of drugs. An ontology based analysis has been conducted, and the individuated cellular processes have been analyzed with gene networks, examining the interactions among genes. A clustering analysis revealed mechanisms shared with other drugs. Results: The results indicate that this compounds have side effects that include inflammatory response and fever, induced by the interleukin signaling pathway. Other evidences related with known effects of the compounds were highlighted. Conclusions: The results indicate that the direct effects could be reached at a post-transcriptional level of P-gp or through other targets, further studies will address these hypothesis. The prediction of side effects will be useful in subsequent *in vivo* experiments.

1 Introduction

New therapeutic strategies tend to be target specific for a personalized treatment. Therefore, confirmation that a compound inhibits the intended target (drug target

validation) and the identification of undesirable secondary effects are the main challenges in developing new drugs. Comprehensive methods that enable researchers to determine which genes or activities are affected by a given drug might improve the efficiency of the drug discovery process by quickly identifying potential protein targets, or by accelerating the identification of compounds likely to be toxic.[3]

The aim of this study is to determine possible targets of three structurally related molecules. The first, PB28, was synthesized as a σ -2 receptor agonist and exhibited an anti-tumoral activity; the other two, MC18 and MC70, derived from PB28. PB28 inhibits cell growth increasing G0-G1 phase cells accumulation and induces caspase-independent apoptosis. PB28 also reduces P-gp expression in a concentration- and time-dependent manner. Human P-gp (MDR1) is an ATP-dependent efflux pump that is capable of transporting many drugs across cell membranes. High level expression of this protein has been linked to tumor resistance to chemotherapy. Our previous studies also showed that other mechanisms may be implied in the effects on cell growth [1]. Subsequently new molecules such as MC70 and MC18 have been synthesized in order to improve the inhibitory effect on P-gp. The present paper further investigate the possible targets of PB28, MC70 and MC18. A genome-wide investigation has been conducted on the mechanism of action of PB28, MC70 and MC18, to explore all potentially affected pathways. The microarray approach has been demonstrated to be useful for unveiling genes physiologically or biochemically relevant to the drug's mode of action and genes related to its resistance if they are regulated at the transcriptional level. For this study, we adopted the Affymetrix GeneChip® system, which provides a reliable data standard for monitoring gene expression, compared with alternative systems [2].

2 Materials and Methods

2.1 Sample Preparation and Microarray Hybridization

An in vitro model constituted by MCF7 ADR cells, breast cancer cell line resistant to adriamycin/doxorubicin (ADR), has been used. The cells were incubated alone or in presence of each drugs, according to the following scheme:

1. MCF7 ADR control
2. MCF7 ADR + PB28 25nM 2D
3. MCF7 ADR + MC18 20mM 2D
4. MCF7 ADR + MC70 20mM 2D

RNA was extracted from each sample by the QIAGEN kit and used for microarray analysis based on the Affymetrix GeneChip Human Gene 1.0 ST system. This chip contains 28.869 genes, each one represented by 26 probes. All the experiments have been conducted in triplicate. Expression levels have been normalized by PLIER.

2.2 Differential Expression Analysis

Differential expression analysis is necessary for individuating genes that constitute an expression pattern of the compound mode of action. Hence the expression of the treated cells is compared to that of the control cells.

Numerous feature selection methods have been applied to the identification of differentially expressed genes in microarray data. These include simple *fold change*, classical *t-statistic* and *moderated t-statistics*. Different methods produce gene lists that are strikingly different.[4]

At low numbers of samples it is difficult to report any differences between methods such as BGA and fold which do not model the variance, and SAM which attempts to model the variance. Equally, in data sets with high variance, fold or non-parametric methods were more powerful than parametric methods.[5] On the other hand, since the number of replicates in many studies is small, variance estimators computed solely within genes are not reliable in that very small values can occur just by chance. Hence a SAM analysis [17] have been performed, calculating a permutation based FDR, and the results have been compared with simple fold-change scores, in order to avoid false positives.

2.3 Ontology-Based Analysis

Because most biological processes involve the complex interaction and regulation of multiple genes, identifying differentially expressed sets of genes has important advantages over identifying individual genes. Many genes may individually exhibit marginal differential expression but may have a significant combined effect on phenotypic outcome [6]. The most common gene set method directly extends single-gene approaches by: (1) Ordering the complete list of genes, L, according to their evidence for differential expression. (2) Examining the occurrence of a predefined gene set S to determine whether it is overrepresented in the top portion of the list, B, relative to the complete list L. (3) Computing a P-value usually based on the Fisher's exact test or its large-sample approximation χ_2 test. In order to individuate the cellular processes affected by the compounds we used the GOMiner [12] Tool and Gene Ontology as gene sets database. We considered for the analyses a list of genes with a statistical significance: $p \leq 0.001$.

2.4 Gene Network Analysis

Efforts to define the interactions among genes and their products yielded a large number of methods to infer gene regulatory network.[8][9] This kind of methods need a huge number of time series or steady state data to infer all the interactions. Some methods that address the individuation of the directly affected genes, have been applied to relatively simple organisms, like in *yeast*,[7] but not yet to complex mammalian cells, as in our case. To perform the network analysis we used the *GenMAPP* software [10][11]. *GenMAPP* is a free computer application designed to visualize gene expression and other genomic data on maps representing biological pathways and groupings of genes. The maps utilized were the ones obtained from the Gene Ontology database, and some other map from the *GenMAPP Contributed MAPP archive*.

2.5 Clustering Analysis

A clustering analysis aimed at finding compounds that act in a similar way to our compounds. An ANOVA filtering individuated the genes differentially expressed

among a database of gene expression levels obtained from microarray analysis of hundreds of compounds, i.e. the Connectivity Map Dataset [15]. The genes survived from the integration among different platforms and variance analysis was used as an input for a SOM analysis. The aim of a SOM network is to find a mapping from the space of dimension equal to the number of components of the data vectors to a one- or two-dimensional space. The mapping should preserve "closeness" between data vectors; that is, two data vectors that are close to one another in the original space should be mapped to points (codebook vectors) of the new space that are also close to one another. For each cluster a prototype was individuated, being the most representative compound for that cluster, and a reliability metric was calculated by means of sub-sampling techniques [18][19][20]. After that clusters are obtained from the analysis of the compounds database, the samples data related to the new compounds are showed to the net, and the distance metric give us a quantification of the membership of the sample to the cluster, and a mode of action closeness as well. These scores are obtained making use of a random-projection based procedure [16].

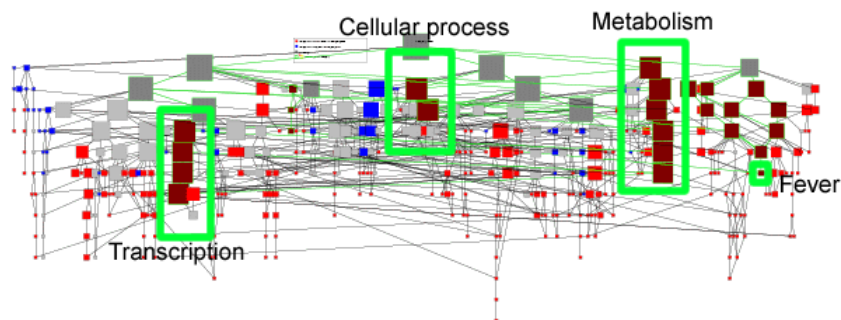
3 Results

Data coming from the differential expression analysis showed that the three compounds present different expression patterns. Genes with a differential expression greater than 80% with respect to gene expression in the controls can be considered differentially expressed. MC18 and MC70 share more similar expression patterns, for the differential expression intensity and for the resulting genes as well. The genes individuated for PB28 principally are chemokine (CCL20) and interleukin (IL1A), both involved in the inflammatory response of the cell; while RNU5E is a gene that may have a role in the alternative splicing of P-gp [14]. MC18 also counts many genes involved in the immune response, like CCL20, CXCL1, C3, IL1A, SERPINA1, IL1B, IFIT2, CFB; TNFAIP3, tumor necrosis factor, alpha-induced protein 3, is a gene that take part of the I-kappaB_kinase/NF-kappaB_cascade activation. PLAU is a gene activated from cell stress. PI3 (peptidase inhibitor 3) is up-regulated by 'alarm signals' such as bacterial lipopolysaccharides, and cytokines such as interleukin-1 and tumor necrosis factor [13]. MC70 shared many genes with MC18, but included some gene over-expressed in PB28 (Table 1) where all the values have an understood positive sign. The genes that can be considered over-expressed (expression greater than 80% of the controls) are bolded in the list.

Many GO terms underlined from the Gene Ontology based analysis, like *transcription* and *regulation of metabolism* in the PB28, are highlighted because they are the terms connecting a specific term (*fever*) to the root. Therefore an interpretation process, exploring the Gene Ontology tree (Fig 1) is needed. This interpretation lead to the elimination of non relevant results; in this case the only surviving elements will remain *fever/heat generation*. With this kind of interpretation, the relevant modulated processes are: *fever* for PB28, *inflammatory response*, *apoptosis*, *nucleosome assembly* and *substrate-bound migration* for MC18 and *inflammatory response*, *chemotaxis*, *apoptosis*, *vasculature development* for MC70. For this kind of analysis genes with a

Table 1. Gene expression patterns of the three compounds coming from the differential expression analysis. Fold change values are reported.

PB28		MC18		MC70	
RNU5E	2.0222	C3	4.0719	PI3	3.1993
CCL20	1.9639	PI3	2.5157	SERPINB2	2.7564
ZNF204	1.8905	IL1A	2.3215	CCL20	2.6144
SNORA62	1.6698	BIRC3	2.2196	IL13RA2	2.4256
hCG_2042718	1.6386	SERPINA1	2.2142	LAMB3	2.3354
ANAPC10	1.6225	PLAU	2.0496	TNFAIP3	2.2345
SNORD45A	1.6093	TNFAIP3	2.0477	IL1B	2.2116
ATRX	1.607	AHNAK2	2.0347	EGR1	2.1702
SNORD25	1.5616	TPCN1	2.0177	NAV3	2.1605
KIAA1641	1.5538	CCL20	1.9805	ESM1	2.126
CEP170	1.5478	ZNF114	1.8712	C3	2.1152
LOC162632	1.5456	IL1B	1.8517	LAMC2	2.1074
EGR1	1.5349	CFB	1.8447	RNU5E	2.049
KIAA1641	1.5349	CFB	1.7916	CXCL1	2.0072
DHFR	1.5315	HSPG2	1.7773	IL1A	1.9769
PWCR1	1.5287	IFIT2	1.7601	FLNC	1.8816
IL1A	1.5166	CA8	1.7527	LOX	1.8695
ZNF43	1.5146	CFB	1.7503	AREG	1.8247
NBR1	1.499	ZNF204	1.7354	IL8	1.7691
SUCLG2	1.4964	LAMC2	1.734	ANTXR2	1.7572

**Fig. 1.** Fever is connected to the root of the tree through the highlighted terms, hence they come out in the results. In a similar way results for the other compounds can be explained.

differential expression of at least 50% have been used. A further analysis was conducted counting the number of genes from the list of genes differentially expressed >50% present in each pathway from BioCarta. Interleukine, cytokine, NFkB and inflammatory pathways resulted to be preponderant. From the gene network analysis of the three compounds using all the gene expression values, TNF-NFkB and Interleukine-1 pathways have been found the only modulated. After filtering genes for platform integration and differential expression, several sample of well known compounds were clustered by the SOM network.

Table 2. Results of the cluster analysis suggest that the three compounds have MOA similarities with Tretinoin and Estradiol

	Bari Compounds					
Clusters	MC18	MC70	PB8	Prototypes	Entropy	Cluster Reliability %
1	0.25333	0.25333	0.248		0.9576	30.0089
5	0.104348	0.10435	0.102609		0.9575	25.0132
8	0.046364	0.04636	0.058182	Valproic Acid	0.955	20.9587
11	0.146667	0.14667	0.10667	Metformin	0.9881	30.3333
12	0.3232	0.3232	0.3408	Estradiol	0.9741	27.3254
13	0.176	0.176	0.136	Tretinoin	0	68.48
14	0.017778	0.01778	0.017778		0.9167	35.9506
15	0.136	0.136	0.136		1	31.36
17	0.08	0.08	0.08		1	16
18	0.002	0.002	0.012		0.9686	32.42
20	0.01	0.01	0.01		0.9788	24.9444

For each cluster a prototype has been individuated. Our compounds have been clustered within the clusters having the Tretinoin and the Estradiol as prototypes (Table 2), since they exhibit the best trade-off between pertinence of the new compounds to the cluster versus cluster reliability. In Table 4 the Entropy of a cluster tell us how much its composition, in terms of compounds, is heterogeneous. The column 2,3 and 4 contain the membership quantification of the compounds respect to a cluster.

4 Discussion

The expression patterns from the three compounds count many genes involved in the inflammatory response, and others like RNU5E that leaves open questions. Genes regulating cell growth were found and nothing let us think at a transcriptional modulation of P-gp leading to suppose that MDR1 is modified at a post-transcriptional level. Looking at a possible list of genes interacting with MDR1 from the *Gene Sorter* tool of the UCSC Genome Bioinformatics Site, that shows expression, homology and other information on groups of genes that can be interrelated, we did not find any gene present in the compounds expression pattern. The different methods used for the differential expression analysis lead to quite identical results. Both indicate an effect of the compounds on inflammatory pathway, and open new questions about the direct target. An in-depth analysis revealed that the cellular processes like *cell death/apoptosis*, *chemotaxis*, *vasculature development*, emerged from the Gene Ontology analysis for MC18 and MC70 because these gene sets contain overlapping genes with the *inflammatory response* process, which is confirmed by the gene network analysis. In fact, the pathways of TNF_NFkB and the Interleukine signaling pathway resulted to be modulated in their wholeness. *Nucleosome assembly* and *substrate-bound migration* are probably due to noise in the data, as a matter of fact these terms contains few genes poorly modulated. The *fever/heat generation* processes resulting for PB28, consist in the interleukine genes, hence correspond to the inflammatory response. This interpretation of the Gene Ontology analysis lead to consider the *inflammatory response* as the only candidate process to be modulated at a transcriptional level for the three compounds. These results are confirmed by the SOM

analysis. Tretinoin is used to treat at least one form of cancer (acute promyelocytic leukemia (APML), usually together with other drugs, by causing the immature blood cells to differentiate (i.e. mature) and die. There is a unique complication of retinoic acid syndrome in patients with acute promyelocytic leukemia. This is associated with the development of dyspnea, *fever*, weight gain, peripheral edema and is treated with dexamethasone. The side effect of fever correspond to the characteristics emerged from the other analysis. Estradiol (17 β -estradiol) (also oestradiol) is a sex hormone. Mislabeled the "female" hormone, it is also present in males; it represents the major estrogen in humans. Estrogen affects certain blood vessels. Improvement in arterial blood flow has been demonstrated in coronary arteries. This is interesting if compared with some modulated processes in MC70, which includes vasculature development. In definitive the results from the different methods seem to validate each other, predicting a possible side effect of the compounds, but not revealing us a clear mechanism of action.

5 Conclusions

The different methods used in this work lead to agreeing results and open new investigation perspectives, opening insights from different points of view. The differential expression analysis are intended to find the principal affected genes, the ontology analysis individuated the cell processes affected through the mediation of the direct targets, the gene network analysis aim to elucidate the interaction between genes, and the SOM analysis gives interesting suggestion about similar compounds. It is evident the effect of all the studied compounds on the immune system of the cell, that was interpreted as a side effect, while further possible targets need a more accurate analysis. All the results presented will be molecularly validated through a RT-PCR analysis.

References

1. Amalia, A., Nicola, A., Colabufo, F.B., Letizia, P., Niso, M., Simone, G.M., Roberto, P., Angelo, P.: Cyclohexylpiperazine Derivative PB28, a s2 Agonist and s1 Antagonist Receptor, Inhibits Cell Growth, Modulates P-glycoprotein, and Synergizes with Anthracyclines in Breast Cancer. *Mol. Cancer Ther.* 5(7), 1807–1816 (2006)
2. Li, J., Pankratz, M., Johnson, J.A.: Differential Gene Expression Patterns Revealed by Oligonucleotide Versus Long cDNA Arrays. *Toxicol. Sci.* 69(2), 383–390 (2002)
3. Marton, et al.: Drug Target Validation and Identification of Secondary Drug Target Effects Using DNA Microarrays. *Nature Medicine* 4(11), 1293–1301 (1998)
4. Pan, W.: A Comparative Review of Statistical Methods for Discovering Differentially Expressed Genes in Replicated Microarray Experiments. *Bioinformatics* 18(4), 546–554 (2002)
5. Jeffery, I.B., Higgins, D.G., Culhane, A.C.: Comparison and Evaluation of Methods for Generating Differentially Expressed Gene Lists from Microarray Data. *BMC Bioinformatics* 7, 359 (2006)
6. Ye, C., Eskin, E.: Discovering Tightly Regulated and Differentially Expressed Gene Sets in Whole Genome Expression Data. In: ECCB 2006, vol. 23, pp. 84–90 (2006)

7. Bernardo, D.D., et al.: Chemogenomic Profiling on a Genome-wide Scale Using Reverse-engineered Gene Networks. *Nature Biotechnology* 23(3), 377–383 (2005)
8. Timothy, S., Gardner, Faith, J.J.: Reverse-engineering Transcription Control Networks. *Physics of Life Reviews* 2, 65–88 (2005)
9. Markowitz, F., Spang, R.: Inferring Cellular Networks: a Review. *BMC Bioinformatics* 8, S5 (2007)
10. Dahlquist, K.D., Salomonis, N., Vranizan, K., Lawlor, S.C., Conklin, B.R.: GenMAPP, a New Tool for Viewing and Analyzing Microarray Data on Biological Pathways. *Nat. Genet.* 31, 19–20 (2002)
11. Doniger, S.W., Salomonis, N., Dahlquist, K.D., Vranizan, K., Lawlor, S.C., Conklin, B.R.: MAPPFinder: Using Gene Ontology and GenMAPP to create a Global Gene-Expression Profile from Microarray Data. *Genome. Biol.* 4, R7 (2003)
12. Zeeberg, et al.: GoMiner: A Resource for Biological Interpretation of Genomic and Proteomic Data. *Genome Biology* 4(4), R28 (2003)
13. Sallenave, J.M., Res, R.: The Role of Secretory Leukocyte Proteinase Inhibitor and Elafin (Elastase-specific Inhibitor/skin-derived Antileukoprotease) as Alarm Antiproteases. *Inflammatory Lung Disease* 1(2), 87–92 (2000)
14. Sontheimer, E.J., Steitz, J.A.: Three Novel Functional Variants of Human U5 Small Nuclear RNA. *Mol. Cell. Biol.* 12(2), 734–746 (1992)
15. Lamb, J., et al.: The Connectivity Map: Using Gene-expression Signatures to Connect Small Molecules, Genes and Diseases. *Science* 313, 1929–1935 (2006)
16. Bertoni, et al.: Random Projections for Assessing Gene Expression Cluster Stability. In: *IJCNN 2005. LNCS*, vol. 3931, pp. 31–37 (2005)
17. Tusher, et al.: Significance Analysis of Microarrays Applied to the Ionizing Radiation Response. *PNAS* 98(9), 5116–5121 (2001)
18. Ciaramella, et al.: Interactive Data Analysis and Clustering of Genomic Data. *Neural Networks* 21(2-3), 368–378 (2007)
19. Ciaramella, et al.: Clustering, Assessment and Validation: an Application to gene Expression Data. In: *Proceedings of the International Joint Conference on Neural Networks. IJCNN 2007*, p. 1419 (2007)
20. Iorio, F., Miele, G., Napolitano, F., Raiconi, G., Tagliaferri, R.: An Interactive Tool for Data Visualization and Clustering. In: Apolloni, B., Howlett, R.J., Jain, L. (eds.) *KES 2007, Part III. LNCS (LNAI)*, vol. 4694, pp. 870–877. Springer, Heidelberg (2007)

Locally Linear Discriminant Embedding for Tumor Classification

Chun-Hou Zheng^{1,2}, Bo Li², Lei Zhang^{3,*}, and Hong-Qiang Wang²

¹ College of Information and Communication Technology, Qufu Normal University, Rizhao, Shandong, 276826 China

Zhengch99@126.com

² Intelligent Computing Lab, Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei, Anhui 230031, China

³ Biometric Research Center, Dept. of Computing, Hong Kong Polytechnic University, Hong Kong, China

cs1zhang@comp.polyu.edu.hk

Abstract. In this paper, followed the assumption that the gene expression data of tumor may be sampled from the data with a probability distribution on a sub-manifold of ambient space, a supervised version of locally linear embedding (LLE), named locally linear discriminant embedding (LLDE), is proposed for tumor classification. In the proposed algorithm, we construct a vector translation and distance rescaling model to enhance the recognition ability of the original LLE from two aspects. To validate the efficiency, the proposed method is applied to classify two different DNA microarray datasets. The prediction results show that our method is efficient and feasible.

Keywords: Manifold learning; Gene expression data; Locally linear embedding; Locally linear discriminant embedding.

1 Introduction

Cancer classification is the critical basis for patient-tailored therapy. With the development of technology, the molecular diagnosis offers a systematic and precise prospect for tumor classification. At present, one of the typical molecular diagnosis methods is DNA microarray technology, which is a powerful tool in functional genome studies [2,3,4,5,8]. This technique has been termed as “class prediction” in the microarray literature [4,9]. By monitoring the expression levels in cells for thousands of genes simultaneously, microarray experiments may lead to a more complete understanding of the molecular variations among tumors, and hence to a finer and more reliable classification.

Gene expression data from DNA microarray are characterized by many measured variables (genes) on only a few observations (experiments), although both the number of experiments and genes per experiment are growing rapidly. In statistical terms, the

* Corresponding author.

very large number of predictors or variables (genes) compared to small number of samples or observations (experiments) make most of classical “class prediction” methods unemployable. Fortunately, this problem can be avoided by selecting only the relevant features or extracting new features containing the maximal information about the class label from the original data. The former methodology is called feature selection or subset selection, while the latter is named feature extraction. In this paper, we will study the nonlinear feature extraction method based on manifold learning.

2 Manifold Learning

2.1 Principle of Manifold Learning for Gene Expression Data

Most manifold learning algorithms assume that the input data resides on or close to a low dimensional manifold embedded in the ambient space. For most applications, the data is generated by continuously varying a set of parameters. Often the number of parameters is not very large. Otherwise, there will not be much benefit from dimensionality reduction. For example, a set of face images can be captured by varying the face pose, scale, position, and lighting conditions. With manifold learning, one can mapping the high dimension observe data to low dimensional manifold.

For gene expression data, we know that the cell state is concern to the environment and its own genes, so they should be determined by these outer and inner factors. Till now people still not clearly realize these factors. If we take the complicated environmental influence and inner gene factors as a set of parameters, then the gene expression data may be resides on or close to a low dimensional manifold embedded in the ambient space with these parameters. According to this assumption, the inherent nonlinear structure hidden in the observe gene expression data may be achieved by using manifold learning. Farther, these low dimensional manifolds may be used for tumor classification, gene regulatory networks construction, etc. In this paper, as an attempt in gene expression data analysis area, we will focus on the tumor classification using manifold learning.

2.2 Locally Linear Discriminant Embedding

2.2.1 The Goal of LLDE

For visualization, the goal of dimensionality reduction methods is to map the original data set into a (2-D or 3-D) space that preserves the intrinsic structure as well as possible. But for classification, it aims to project the data into a feature space in which the members from different classes could be clearly separated. LLE is an effective dimensionality reduction approach to visualize the high dimensional data into a 2-D space. However, little classification ability can be displayed by implementing the original LLE.

Based on the fact mentioned above, in this paper, we propose a supervised LLE algorithm, which named as locally linear discriminant embedding (LLDE). The goal of LLDE is to take full advantage of the class information to improve the classification ability of the original LLE. It is well known that the reconstructing weights are invariant to translation under sum-to-one constraint in the original LLE, which can be confirmed by :

$$\Phi(Y) = \sum_i \left\| Y_i - \sum_j W_{ij} Y_j \right\|^2 = \sum_i \left\| (Y_i - T_i) - \sum_j W_{ij} (Y_j - T_j) \right\|^2 \tag{1}$$

where T_i is a translation vector corresponding to class i . Thus each point with the same class can be translated by the same vector T_i and so does to the points belonging to different labels with the corresponding vectors.

In order to overcome the out-of-sample problem, a linear transformation, i.e. $Y = A^T X$, is plug. Thus the objective function of the original LLE can be changed into the following form.

$$J_1(A) = \min tr\{YMY^T\} = \min tr\{A^T XMX^T A\} \tag{2}$$

Some studies have found that a linear version of LLE shows better recognition ability than the original LLE [6]. However, the linear transformation is not always the optimal one that the proposed LLDE is pursuing for. That is to say, LLDE needs a criterion which can automatically find an optimal linear transformation for classification.

In this paper, a maximizing margin criterion (MMC)[7]was proposed to determine the optimized subspace, which can successfully conquer the small sample size problem. The objective function is stated below:

$$J_2 = \max \left\{ \sum_{ij} p_i p_j \left(d(m_i, m_j) - s(m_i) - s(m_j) \right) \right\} \tag{3}$$

where p_i and p_j are the prior probability of class i and class j , m_i and m_j are centroids of class i and class j . $d(m_i, m_j)$, $s(m_i)$ and $s(m_j)$ express the distance of centroids and their trace, respectively. So the optimized function can be deduced to

$$J_2 = 2 \max tr(S_b - S_w) \tag{4}$$

In order to take advantage of the property of the weighted matrix's invariance to rescaling, the rescaling parameter μ is introduced to Eqn. (4). Here, we first translated the data to suitable places, then rescaled the data with the same label to their centroids and all the centroids were kept unchanged. So S_b was still preserved and S_w was rescaled. And the objective function can be deduced to :

$$J_3 = \max tr(S_b - \mu S_w) \tag{5}$$

Where μ denotes the rescaling coefficient and $\mu > 0$. Then the modified maximizing margin criterion (MMMC) can be obtained and rewritten to the following form.

If a linear transformation $Y = U^T X$ can maximize Eqn. (5), then an optimal subspace for classification will be explored. This is because the linear transformation aims to project a pattern closer to those with the same class but farther from patterns in different labels, which is just the goal for classification. Under such circumstance, the distance between different centroids will be larger and the within class scatters will be smaller. In other words, $d(m_i, m_j)$ will be maximized and $s(m_i)$ will be

minimized simultaneously in this subspace. Thus $\sum_{ij} p_i p_j (d(m_i, m_j) - s(m_i) - s(m_j))$ and $tr(S_b - \mu S_w)$ will be maximized accordingly. That is to say, to find an optimal subspace for classification means to maximize the following optimized function.

$$J_2 = tr\{U^T (S_b - \mu S_w) U\} \tag{6}$$

2.2.2 Discriminant Feature Extraction

Let $T_{d \times n}$ denote the translation matrix, thus the discriminant component after performing LLDE can be represented to $[Y - T]_{d \times n}$, which can also be represented by a linear transformation, i.e. $Y - T = V^T X$.

Based on the analysis mentioned above, it can be found that the linear approximation to the original LLE explores a linear subspace with the least reconstructed error. Moreover, under the sum-to-one constraint, this linear transformation can translate the points to random places, which impacts on the recognition rate of the data greatly. In other words, the linear approximation to LLE can improve the discriminability of the data. However, the projection can not be ensured optimal. At the same time, a MMMC is presented above, which can map the data into an optimal subspace for classification. That is to say, if the linear transformation obtained by linearized LLE can satisfy Eqn. (6), the discriminability of the data will be improved greatly. Thus the problem can be represented as the following multi-objects optimized problem.

$$\begin{cases} \min tr\{V^T X M X^T V\} \\ \max tr\{V^T (S_b - \mu S_w) V\} \end{cases} \tag{7}$$

Moreover, there is one constraint in LLE, that is

$$V^T X X^T V = nI \tag{8}$$

The constrained multi-objects optimized function has intent on minimize the reconstructed error and maximize the margin between difference classes simultaneously. So it can be changed into the following constrained problem.

$$\begin{aligned} &\min tr\{V^T (X M X^T - (S_b - \mu S_w)) V\} \\ &s.t. \quad V^T X X^T V = nI \end{aligned} \tag{9}$$

We use the Lagrangian multiplier to solve the above optimization problem and get Thus we can get

$$(X M X^T - (S_b - \mu S_w)) V = \lambda X X^T V \tag{10}$$

where λ_i is the generalized eigenvalue of $(X M X^T - (S_b - \mu S_w))$ and $X X^T$, V_i is the corresponding eigenvector. Therefore, the objective function is maximized when V is composed of the first d smallest eigenvectors of the above generalized eigen-decomposition.

3 Experimental Results

In this section, the performance of LLDE is evaluated on two experiments and compared with some other methods. The number of nearest neighbors for constructing the nearest neighbor graph in the proposed algorithm was set to 5. After extracting features using the proposed algorithm, different pattern classifiers can be adopted for recognition, including k -NN, neural networks, support vector machine (SVM), etc. In this study, we apply the 1-NN classifier for its simplicity. The Euclidean metric is used as our distance measure. Two publicly available microarray datasets are used to study the tumor classification problem. They are acute leukemia data [4], and High-grade glioma data [10], respectively. In these datasets, all data samples have already been assigned to a training set or test set. An overview of the characteristics of the two datasets can be found in Table 1.

Table 1. Summary of the datasets for the two binary cancer classification problems

Datasets	Training set		Test set		Genes
	Class 1	Class 2	Class 1	Class 2	
Acute leukemia data	11	27	14	20	7129
High-grade glioma data	21	14	14	15	12625

To obtain reliable experimental results showing comparability and repeatability for different numerical experiments, this study not only uses the original division of each data set in training and test set, but also reshuffles all datasets randomly. In other words, all numerical experiments were performed with 20 random splitting of the four original datasets. And, they are also stratified, which means that each randomized training and test set contains the same amount of samples of each class compared to the original training and test set. We built the classification models using the training samples, and estimated the classification correct rates using the test set. The best classification results for tumor and normal tissues using our proposed methods (LLDE+KNN) are displayed in Table 2. For each classification problem, the experimental results gave the statistical means and standard deviations of the accuracy on the original data set and 20 randomizations as described above, with the dimension listed in Table 2. Since the random splits for training and test set are disjoint, the results given in Table 2 should be unbiased.

To show the efficiency and feasibility of the method proposed in this paper, the results using other three methods (Methods 1-3) are also listed in Table 2 for comparison. Considering the ‘high dimension and little sample’ character of gene expression, SVM may be the best classifier for classifying the original data[3]. During the experiment, we trained the SVM with RBF kernel using leave one out cross validation (LOO-CV) performance on training set and then using it to classify the test data. In method 2, we first use the conventional feature extraction technique, i.e. PCA to reduce the dimensions of the gene expression data, then use k -NN for classification.

Table 2. The classification accuracy on test set

No.	Methods	Leukemia data		Glioma data	
		Accuracy	Dimension	Accuracy	Dimension
1	SVM	93.56±4.12		66.55±4.00	
2	PCA+KNN	80.29±7.34	9	73.79±6.13	6
3	SLLE +KNN	92.65±4.44	6	73.79±5.68	12
4	LLDE+KNN	95.59±2.86	3	75.17±7.76	17

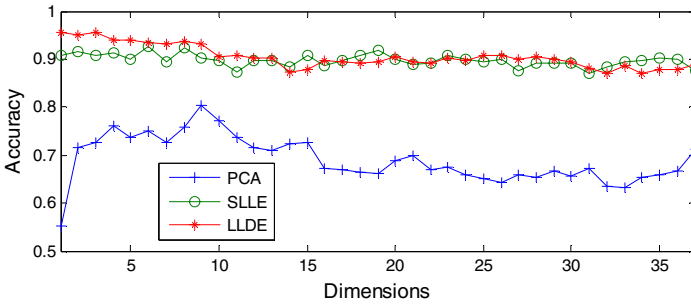


Fig. 1. The mean accuracy on the test set of Leukemia data

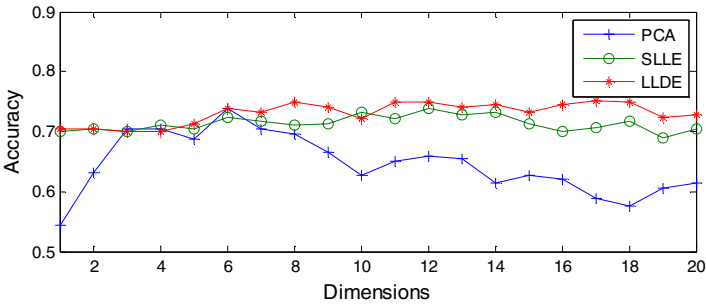


Fig. 2. The mean accuracy on the test set of Glioma data

Supervised LLE (SLLE) [1,12] is a variation on LLE for supervised classification problems, which has been successfully used for gene expression data classifying [11]. In these two experiments, the number of nearest neighbors for constructing the nearest neighbor graph of SLLE was set to 5. All the experimental results of the three methods listed in Table 2 are also the best classification results with the statistical means and standard deviations of the accuracy on the original test data set and 20 randomizations splits.

To roundly show the experimental results, we show the accuracies of Methods 2 to 4 in Figure 1 and Figure 2 when the dimensions change from 1 to the number of the

training sample subtract from 1, in which, every Accuracy is the mean of 21 experiments with a certain dimension.

From Table 2 and Figures 1 and 2 it can be seen that, for the two datasets, both SLLE and DLLE could achieve satisfactory result. SVM can classify the Leukemia data very well, yet it is the worst method for classifying Glioma data. In summary, our method is more competitive both in the mean accuracy and the standard deviation.

During the experiment, we have also tested the rescaling coefficient's impact on the recognition rate, yet the experimental results show that the rescaling coefficient shows few effects on the recognition, so we have not shown the change of the recognition rate when the rescaling coefficient μ varying.

4 Conclusions

In this paper, we presented a discriminant method based on the classical LLE and studied its application in gene expression data. The proposed approach can effectively extract the most discriminant features. Compared with other feature extraction algorithms, the new technique does not suffer from the small sample size problem, the problem of dimensionality reduction for two times and the disconnected components problems. The experimental results show that the new method is effective.

In fact, tumor should be concerned with a lot of factors. Since we don't know these factors and how these factors affect the gene expression now, so the manifold learning may be a good choose to analyse the gene expression data. Our work in this paper is an attempt in manifold and bioinformatics fields, in the future, more works should be done to deep validate whether or not the study is rational.

Acknowledgements. This work was supported by the grants of the National Science Foundation of China, No. 30700161, the grant of the Guide Project of Innovative Base of Chinese Academy of Sciences (CAS), No.KSCX1-YW-R-30.

References

1. Bai, X.M., Yin, B.C., Shi, Q., Sun, Y.F.: Face Recognition Based on Supervised Locally Linear Embedding Method. *J. Inform. Comput. Sci.* 4, 641–646 (2005)
2. Cordero, F., Botta, M., Calogero, R.A.: Microarray Data Analysis and Mining Approaches. *Briefings in Functional Genomics and Proteomics* 6(4), 265–281 (2007)
3. Furey, T.S., Cristianini, N., Duffy, N., Bednarski, D.W., Schummer, M., Haussler, D.: Support vector Machines Classification and Validation of Cancer Tissue Samples Using Microarray Expression Data. *Bioinform* 16, 906–914 (2000)
4. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S.: Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. *Science* 286, 531–537 (1999)
5. Huang, D.S., Zheng, C.H.: Independent Component Analysis Based Penalized Discriminant Method for Tumor Classification Using Gene Expression Data. *Bioinformatics* 22(15), 1855–1862 (2006)

6. Kokiopoulou, E., Saad, Y.: Orthogonal Neighborhood Preserving Projections. In: Proceedings of the Fifth IEEE international Conference on Data Mining, pp. 1–7 (2005)
7. Li, H.F., Jiang, T., Zhang, K.S.: Efficient and Robust Feature Extraction by Maximum Margin Criterion. *IEEE transaction on neural networks* 17(1), 157–165 (2006)
8. Liao, J.G., Chin, K.V.: Logistic Regression for Disease Classification Using Microarray Data: Model Selection in A Large P and Small N Case. *Bioinformatics* 23(15), 1945–1951 (2007)
9. Nguyen, D.V., Roche, D.M.: Tumor Classification by Partial Least Squares Using Microarray Gene Expression Data. *Bioinformatics* 18(1), 39–50 (2002)
10. Nutt, C.L., Mani, D.R., Betensky, R.A., Tamayo, P., Cairncross, J.G., Ladd, C., Pohl, U., Hartmann, C., McLaughlin, M.E., et al.: Gene Expression-Based Classification of Malignant Gliomas Correlates Better with Survival than Histological Classification. *Cancer Res.* 63, 1602–1607 (2003)
11. Pillati, M., Viroli, C.: Supervised Locally Linear Embedding for Classification: An Application to Gene Expression Data Analysis. In: Proceedings of 29th Annual Conference of the of the German Classification Society (GfKI 2005), pp. 15–18 (2005)
12. Ridder, D., Duin, R.P.W.: Locally Linear Embedding for Classification. Technical Report PH-2002-01, Pattern Recognition Group, Department of Imaging Science and Technology, Delft University of Technology, Delft, The Netherlands (2002)

Palmprint Linear Feature Extraction and Identification Based on Ridgelet Transforms and Rough Sets

Shanwen Zhang, Shulin Wang, and Xuelin Li

Hefei Institute of Intelligent Machines, Chinese Academy of Science, Hefei,
Anhui 230031, P.R. China
Zhangshanwen1965@163.com

Abstract. As one of the most important biometrics features, palmprint with many strong points has significant influence on research. In this paper, we propose a novel method of palmprint feature extraction and identification using ridgelet transforms and rough sets. Firstly, the palmprints are first converted into the time-frequency domain image by ridgelet transforms without any further preprocessing such as image enhancement and texture thinning, and then feature extraction vector is conducted. Different features are used to lead a detection table. Then rough set is applied to remove the redundancy of the detection table. By this way, the length of conduction attribute is much shorter than that by traditional algorithm. Finally, the effectiveness of the proposed method is evaluated by the classification accuracy of SVM classifier. The experimental results show that the method has higher recognition rate and faster processing speed.

Keywords: Palmprint feature identification; Ridgelet transforms; Rough sets.

1 Introduction

Personal identification is known as biometrics computing began in 1970s. At that time, fingerprint based automatic checking systems are widely used in law enforce. Retina and iris based systems were introduced in the mid 1980s. Today's speaker identification biometrics have their roots in technological achievements of the 1970s; while signature identification and facial recognition are relative newcomers to the industry. Now worldwide, there are many applications of biometrics being used or considered. Most of the applications are still in the initializing stage of testing and are optional for end users. Any situation that allows an interaction between man and machine is capable of incorporating biometrics. The situations may fall into a range of application areas such as computer desktops, networks, banking, immigration, law enforcement, telecommunication networks and monitoring the time and attendance of staff. Fraud is an ever-increasing problem and security is becoming a necessity in many walks of life. palmprint based identification system has several special advantages such as stable line features, rich texture features, low-resolution imaging, low-cost capturing devices, easy self positioning, and user-friendly interface etc. For these reasons, nowadays the research related to this issue is becoming more active. So far, there have been many

approaches proposed for palmprint recognition including verification and identification[1-3]. Zhang and Kong proposed PalmCode method, which employed one 2D Gabor filter to extract the texture feature of palmprint [4]. Zhang et al. used over-complete wavelet expansion and directional context modeling technique to extract principal lines-like features [5]. And in order to analyze the palmprint images in a multi-resolution-frequency representation, they also adopted wavelet transformation at the same time [6]. Ordinal Code proposed by Z.N. Sun et al. and Robust Line Orientation Code proposed by W. Jia were also using orientation feature for palmprint recognition [7-9]. A. Kumar proposed an approach combining hand shape and palmprint texture features [10].

Linear feature detection is a foremost problem in palmprint image processing and pattern recognition. The palmprint is composed of a number of points, lines and planes. Effective detections of these components play an important role in simplifying feature model in pattern recognition and pattern match. Ridgelet transforms, related to the wavelets and the Radon transform, offers a sound mathematical framework to organize linear information at different scales of resolution, and it performs very well for objects with point singularities two dimensions. However, in palmprint compression and image edge detection, both wavelets and Fourier transforms are far from efficient in extracting edges in images. Edges lie in the regions where the palmprint is discontinuous. So the palmprint data is regarded as a 2-D function, which is smooth away from line-like singularities and singular across edges. For a short straight line in the palmprint, we can use Radon transforms and Hough transforms to obtain it. But for a curve, the detection algorithm is very complicated. Ridgelet[11,12] is just a new analysis tool to get the efficient approximation of images by exhibiting some special sorts of higher-dimensional spatial in homogeneities, and it is developed on the basis of the concepts and methods of modern harmonic analysis and the group representations theory and wavelet analysis, which describes general functions as a superposition of ridge functions in a stable and concrete way. One interesting property of the ridgelet transforms is coefficient sparsely. Higher coefficients are concentrated around the angle θ and situation t corresponding to the longer lines in the image. Thus, the sparsity permits us to localize and to separate linear singularities into the parameter space. This is the main property that distinguishes this wavelet from usual separable wavelets. This paper presents a new method of palmprint identification using ridgelet transforms. Experimental results show that the method is more effective.

2 Relation between Ridgelet Transforms, Wavelet Transforms, and Radon Transforms

The definition of ridgelet is as follows: Denoting a function $\psi : \mathfrak{R} \rightarrow \mathfrak{R}$, satisfying a condition,

$$\int |\hat{\psi}(\xi)|^2 |\xi|^{-2} d\xi < \infty \quad (1)$$

where, ψ is ridgelet base function, $\psi_{a,b,\theta}$ in 2-D is defined from ψ ,

$$\psi_{a,b,\theta}(x) = a^{-1/2}\psi((x_1 \cos \theta + x_2 \sin \theta - b)/a) \tag{2}$$

The continuous 2-D ridgelet transforms (CRT) of a signal $f(x)$ can be written as

$$CRT_f(a,b,\theta) = \int \bar{\psi}_{(a,b,\theta)}(x)f(x)dx \tag{3}$$

The ridgelet inverse transforms as fellow

$$f(x) = \int_0^{2\pi} \int_{-\infty}^{+\infty} \int_0^{+\infty} \mathfrak{R}_f(a,b,\theta)\psi_{a,b,\theta}(x) \frac{da}{a^3} db \frac{d\theta}{4\pi} \tag{4}$$

For comparison, the continuous 2-D wavelet transforms (CWT) of a signal $f(x)$ can be written as

$$CWT_f(a_1,a_2,b_1,b_2) = \int_{R^2} \psi_{a_1,a_2,b_1,b_2}(x) f(x) dx \tag{5}$$

where the wavelets in 2-D are tensor products

$$\psi_{(a_1,a_2,b_1,b_2,\theta)}(x) = \psi_{(a_1,b_1,\theta)}(x_1) \cdot \psi_{(a_2,b_2,\theta)}(x_2) \tag{6}$$

In 2-D, points and lines are related via radon transforms(RT), thus the wavelet and ridgelet transforms are linked via the radon transforms. More precisely, denote the Radon transforms as

$$RT_f(\theta,t) = \int f(x_1,x_2)\delta(x_1 \cos \theta + x_2 \sin \theta - t)dx_1dx_2 \tag{7}$$

where $(\theta,t) \in [0,2\pi) \times R$.

As can be seen, the CRT is similar to the 2-D CWT except that the *point* parameters (b_1,b_2) are replaced by the *line* parameters (b,θ) . In other words, these 2-D multiscale transforms are related by **Wavelets** $\Rightarrow \psi_{scale,point-position}$, **Ridgelets** $\Rightarrow \psi_{scale,line-position}$ (Seeing the Fig.1).

The ridgelet transforms is the application of 1-D wavelet transforms to the slices of the Radon transforms, while the 2-D Fourier transforms is the application of 1-D Fourier transforms to those Radon slices. So in the two-dimensional case, the information provided by a ridgelet coefficient CRT_f is just the one-dimensional wavelet coefficient of CWT_f , i.e. the radon transforms of RT_f . Hence the ridgelet transforms is

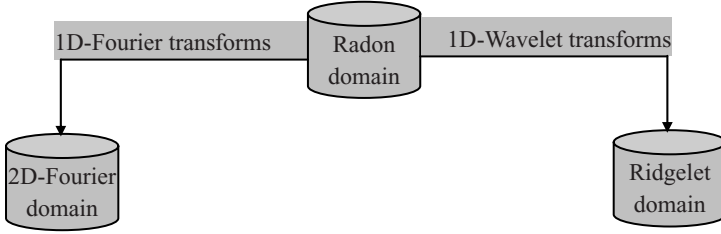


Fig. 1. Relations between transforms

precisely the application of a one-dimensional wavelet transforms to the slices of the Radon transforms of a function, where θ stands for the orientation of Radon transforms and t is a variable parameter.

In fact, although it is an independent image transform, originated in the context of non-separable wavelets, the ridgelet transforms can also be explained as the combination of the Radon transform and the wavelet transforms. This particular point of view relates this work to other recent approaches for shape representation and retrieval based on the Radon/Hough transform application of wavelets to the Radon transform of an image. Therefore, as the Radon space corresponds to the parameters of the lines in the image, and applying wavelets allows to detect singularities, the ridgelet transforms will detect singularities in the Radon space, which will correspond to the parameters of relevant lines in the image. Therefore, the ridgelet transforms combines advantages from both transforms, the ability to detect lines, from the Radon transform, and the multiscale property of wavelets to work at several levels of detail.

3 Attribute Reducing Algorithm of Rough Set

In a data table, it is well known that not all conditional attributes are necessary to depict the decision attribute before decision rules are generated. To acquire brief decision rules from decision systems, knowledge reduction is needed. Knowledge reduction for a database aims to search for some particular subsets of condition attributes that preserves the same properties as the condition attribute set.

We denote $\gamma_B(D) = \frac{|Pos_B(D)|}{|U|}$ the dependency of decision attribute D to condition

attribute subset $B \subseteq C$, where $Pos_B(D)$ is the subset of tumor samples whose neighborhoods consistently belong to one of the decision classes, and $SIG(a, B, D) = \gamma_B(D) - \gamma_{B-a}(D)$ denotes the significance condition attribute a with respect to condition attribute subset $B \subseteq C$. The algorithm is described as follow:

Input: $RD = \langle U, C, D \rangle$

```
// RD is a rough decision table, U denotes tumor sample set
// C is a condition attribute set, D is a attribute set
```

Output: red //reduction of condition attribute set C
 step1: $red = \emptyset$; //Initial red is set to empty set and is the pool to contain the informative attribute.
 step 2: For each $\forall a_i \in C - red$
 Computing $SIG(a_i, red, C) = \gamma(C, red \cup a_i) - \gamma(C, red)$
 // where, define $\gamma(C, \emptyset) = 0$, the dependency of subtype table
 //the set C with respect to empty set is fixed to zero.
 step3: Selecting the informative gene a_k satisfying
 $SIG(a_k, red, C) = \max_i(SIG(a_i, red, C))$;
 step4: if $SIG(a_k, red, C) > 0$
 $red = red \cup a_k$ // Add the optimal gene a_k to the reduction set red
 go to step2;
 else
 return red ;
 step5: Algorithm end;

4 A Method of Palmprint Identification by RT and RS

The question about palmprint identification has complicated nature of incertitude. In many conditions, we always get cursory information in the beginning of investigation. So the key of the problem is how to engender the final rule using useful feature and original data. To deal with the decision information using ridgelet transforms gets feature vector. To express acquired rule uses the form of relation table and logic arithmetic. A method of plamprint identification is as follows:

- 1) To pretreatment the original plamprint images $f(m, n)$ ($m, n = 0, 1, 2, \dots, L-1$), such as mending the miss data and uniting the repeated object.

- 2) To select the ridgelet function $\psi_{j,k,i}(m,n) = a_j^{-1}\psi(2^j(L_{j,i}(m,n) - b_{j,k}))$, and to decompose ridgelet transforms for the pretreatment data, denoting the coefficients $\alpha_{j,k,i} = \langle \psi_{j,k,i}, f(m,n) \rangle$.
- 3) Ordering two thresholds λ, ε , for each k , defining two sets, toning up the plamprint images, $A_k = \{x_{i,j} \mid \alpha_{j,k,i} > \lambda\}$, $B_k = \cup_{i,j} \{x_{i,j} \mid |\alpha_{j,k,i} - \alpha_{j\pm 1,k,i\pm 1}| > \varepsilon\}$.
- 4) Calculate the ridgelet transforms energy of each k , get an eigenvector from the A_k, B_k , form the relation table consisting conditional attribute aggregate and decision attribute aggregate.
- 5) By above attribute reducing algorithm of rough set, to predigest the relation table, delete the redundant attribute and unite repeated object in turn, and predigest each object, delete redundant attribute.
- 6) To calculate the core valued and the possible predigesting of conditional attributes, and make relevant rule.

For identification rules are not alone, we choose attribution table according to definite rule and get the excellent and brief rule finally.

5 Experimental Results and Analysis

The objective of the experiments is to show the difference between attribute reduction with traditional rough set model and covering rough sets. As traditional model can just deal with discrete attributes, we employ two discretization techniques, equal-width, equal-frequency, to transform the numerical data into discrete one. There is a parameter to be specified in using covering rough sets to compute dependency, the size of the covering of an object. We try the size of covering delta at a step of 0.05. Before reduction, all of the numerical attributes are standardized at interval [0,1]. During the experiment, we employ radial basis function(RBF) kernel in support vector machine(SVM) learning algorithm to validate the selected features. The covering decision systems can be divided into consistent covering decision systems and inconsistent decision systems.

The following experiments are designed for testing the accuracy and efficiency of the proposed method. The data collection process involves four steps:

- Find 50 people of different ages, sexes and occupations.
- Capture six palmprint samples from each person.
- Randomly pick up one from the six samples to set up the database.
- Use the left 250 samples (each person has five) as the testing set.

All the palmprints are from right hand and are captured with the same palmprint capture device. Palmprint samples from the same palm may be with a little rotation and shift. The size of all the palmprint images is 240×240.

To ridgletet transform for each palmprint image, obtaining coefficients. To give a threshold T , and order

$$G_{i,k,j} = \begin{cases} 0 & \text{if coefficients} < T \\ 1 & \text{if coefficients} \geq T \end{cases} \quad (8)$$

Then, $G_{i,k,j}$ is lines-image of the palmprint, seeing the Fig.3.

The Fig.2 shows an example of palmprint-line-feature. Fig.2(b) is the palmprint-line-feature for $k=54$, Fig.2(c) is the fusion of palmprint-line-feature for $k=54$ and $k=78$, Fig.2(d) is the fusion of palmprint-line-feature for $k=54$, $k=78$, and $k=126$. From the figure2, all palm lines are represented clearly and accurately.

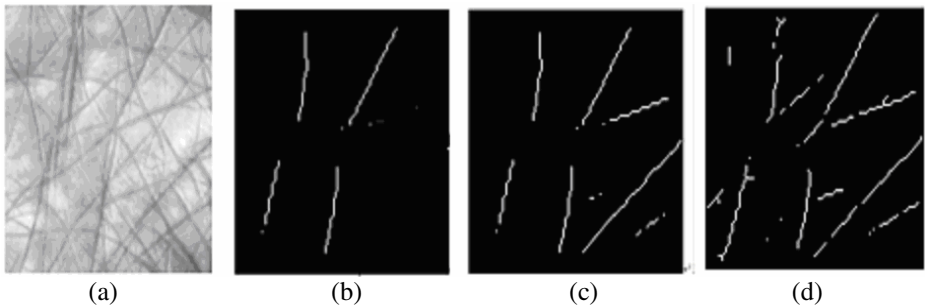


Fig. 2. Images appearing in feature extraction stage: (a) Original image; (b) line-feature image for $k=54$; (c) fusion line-feature image for $k=54$ and $k=78$; (d) fusion line-feature image for $k=54$, $k=78$ $k=126$.

We extract the feature vector from the coefficients, obtaining the relation table, every object in the table has 50 condition attributes. Basing on attribute reduction method of rough set, the number of condition attribute is reduced to 28. In this experiment, we use 30 samples as training data and the other 20 samples as testing data. We try the following two kinds of support vector machines: 1) Linear SVM (no kernel); 2) Radial basis function SVM (RBF kernel).

Where, we select wavelet function Daubechies5. The results can be seen from table1. In order to compare, the palmprint identification based on other methods are given in table.1.

Table 1. Results of palmprint identification rate

Method	PCA and SVM	ICA and SVM	RT,RS and SVM
Identification rate	96%	95%	95%
response time (s)	12	14	8

The response time is recorded to evaluate the efficiency of the proposed method. From the table1, we can see the timing response is shorter using the method which is presented in this paper.

6 Conclusions

Mutual information has recently been proposed for feature selection. But it often contains redundancy in the feature set selected by this method. Attribute reduction in rough set theory provides a feasible way to deal with redundancy and does not reduce the contained information. In this paper, we introduce a method of feature extraction and identification for palmprint retrieval based on ridgelet transforms and rough set. First, the palmprints are first converted into the time-frequency domain image by ridgelet transforms, and then feature extraction vector is conducted. Different features are used to lead a detection table. Then rough set is applied to remove the redundancy of the detection table. Finally, the effectiveness of the proposed method is evaluated by the classification accuracy of SVM classifier. Experiments are carried out in order to measure the performance of the proposed method. Experimental results prove the efficiency of the proposed method.

Acknowledgements. This work was supported by the grants of the National Science Foundation of China, Nos. 60705007 & 60772130.

References

1. Li, W., Zhang, D., Xu, Z.: Palmprint Recognition Based on Fourier Transform. *Journal of Software* 13(5), 879–908 (2002)
2. Jia, W., Huang, D.S., Zhang, D.: Palmprint Verification Based on Robust Line Orientation Code. *Pattern Recognition* 41(5), 1521–1530 (2008)
3. Huang, D.S., Jia, W., Zhang, D.: Palmprint Verification Based on Principal Lines. *Pattern Recognition* 41(4), 1316–1328 (2008)
4. Ramos, T., Valveny, E.: A New Use of the Ridgelets Transform for Describing Linear Singularities in Images. *Pattern Recognition Letters* 27, 587–596 (2006)
5. Zhang, D., Kong, A., You, J., Wong, M.: Online Palmprint Identification. *IEEE Trans. Pattern. Anal. Mach. Intell.* 25(9), 1041–1050 (2003)
6. Zhang, L., Zhang, D.: Characterization of Palmprints by Wavelet Signatures via Directional Context Modeling. *IEEE Transaction on Systems, Man and Cybernetics, Part B* 34(3), 1335–1347 (2004)
7. Connie, T., Jin, A.T.B., et al.: An Automated Palmprint Recognition System. *Image and Vision Computing* 23(5), 501–515 (2005)
8. Sun, Z.N., Tan, T.N., Wang, Y.H., Li, S.Z.: Ordinal Palmprint Representation for Personal Identification. In: *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 279–284 (2005)
9. Jia, W., Huang, D.S., Zhang, D.: Palmprint Verification Based on Robust Line Orientation Code. *Pattern Recognition* 41(5), 1429–1862 (2008)
10. Kumar, Zhang, D.: Personal Recognition Using Hand Shape and Texture. *IEEE Transactions on Image Processing* 15(8), 2454–2461 (2006)
11. Candes, E.: Ridgelets: Theory and Applications, Ph.D. Thesis, Department of Statistics, Stanford University (1998)
12. Candes, E., Donoho, D.L.: Ridgelets: A Key to Higher-Dimensional Intermittency? *Phil. Trans. R. Soc. Lond. A*, 2495–2509 (1999)

A Compact Genetic Algorithm with Elitism and Mutation Applied to Image Recognition

Rafael R. Silva, Heitor S. Lopes*, and Carlos R. Erig Lima

Bioinformatics Laboratory
Federal University of Technology Paraná (UTFPR),
Av. 7 de setembro, 3165 80230-901, Curitiba (PR), Brazil
rafael.rsi@gmail.com, hslopes@pesquisador.cnpq.br, erig@utfpr.edu.br

Abstract. The problem of object recognition in images is a hard problem frequently found in industrial and academic application. This work presents the application of an extension of the Compact Genetic Algorithm (emCGA) to three problems of object recognition in real images. Results are compared with an exhaustive search algorithm and another CGA. Results suggested the efficiency of emCGA for this problem and encourages future developments.

Keywords: Compact Genetic Algorithm; object recognition.

1 Introduction

The Genetic Algorithm (GA) is an efficient tool for optimization problems and has been applied on several engineering problems. However, in some applications, the computational cost of the GA can be too high, demanding a prohibitive execution time or excessive hardware resources. A possible alternative is the use of a genetic algorithm with smaller computational complexity, which can run in less powerful systems. Alternatively, it can be implemented in parallel architectures using reconfigurable logical devices [2]. A possible limitation of the GA implementation is the amount of memory required to store the population. This is particularly true for hardware implementations. A GA evolves a population, not a single point, thus requiring memory space to store such information.

On the other hand, the CGA (Compact Genetic Algorithm) can achieve the same level of quality of a SGA (Simple Genetic Algorithm) with uniform crossover, but using less memory to store the population. This is possible because the CGA works with a probability vector instead of the whole population [5].

Another feature of CGA is the use of techniques to evolve the probability vector, imitating the behavior of a SGA. Due to the simplicity of these techniques and the small memory requirements, some works proposed software and hardware implementations of the CGA, showing good results with significant resources reduction for its construction.

* This work was partially supported by the Brazilian National Research Council – CNPq, under research grant no. 309262/2007-0 to H.S. Lopes.

This work presents the application of a new extension of a CGA, the emCGA [8], to object recognition in images. This approach uses elitism and introduces a new mutation operator. Differently from other mutation operators introduced for CGA, this one is applied to new individual generation, thus imitating the crossover operator of a SGA. This operator does not increase significantly computational cost or memory consumption and increases the overall performance, when compared with similar works.

2 The Compact Genetic Algorithm

The CGA is an Estimation of Distribution Algorithm (EDA), first proposed by Harik et al. [5], that generates descendants using a statistical population model, instead of the traditional recombination operators and mutation. [4] estimated the convergence time for a special class of GA problems without iterations among building blocks (BB). The idea of CGA was to simulate an independent random walk model for each bit of the chromosome. As result, the population is reduced to a vector of probabilities that occupies only $L * \log_2(N)$ bits of memory, in comparison with $L * N$ for a SGA (where L is the chromosome length and N the size of the population). The CGA can imitate the behavior of a SGA with uniform crossover using a reduced amount of memory. The importance of the size of the population in GA performance has been focused in other works, such as [4]. Usually, a large population size results in better quality of the solution, but it increases the computation cost and memory use.

Since CGA was introduced by [5], several extensions were proposed aiming to improve its performance. Their main focus are the introduction of techniques to allow the CGA to overcome the performance reduction in problems with higher order building blocks. The extensions of CGA that proposed some elitist technique were those that showed the best balance between demand of resources and performance. Elitism allows to increase the selective pressure and to reduce the genetic drift.

Despite of the competitive performance of CGA with SGA for low order BBs, it does not achieves the same performance when high order BBs are present in the problem. In this case, with the compact representation of the population in CGA, the information regarding high order relationships between the chromosome bits does not survive throughout generations, differently that what happens in SGA. For real-world problems, these disturbances generated by the uniform crossover should be overcome, since many problems presents local optima (i.e. multimodal) and interdependent genes (high order BB). However, increasing the selective pressure in CGA tends to decrease these noisy effects, because it increases the probability of high order BBs to survive throughout generations [4],[1]. Some works proposed modifications in the basic CGA to improve performance by increasing the selective pressure, such as: neCGA [1], mCGA [3] and emCGA [8]. Among them, the emCGA showed the best tradeoff between solution quality (fitness value) and convergence speed (number of evaluations).

2.1 The emCGA

The previously mentioned works (neCGA and mCGA) perform better than the original CGA. Both works use elitism, but mCGA uses the mutation operator with success. Another work has also proposed a mutation operator [9], called MBBCGA, but it is different from the mCGA mutation operator. This work presents the application of an extension operator, aimed at improving even more the quality of solutions but, also, keeping a reasonable convergence speed. The previously mentioned works that use mutation operators do not focus on population diversity control, but on local search. In the neCGA the mutation is applied to the elite individual of the current generation to generate another individual. After a tournament over these individuals, the best one will be the elite for the next generation. In the mCGA, mutation is applied in the first individual to substitute the second generation of the conventional CGA.

The new proposed operator allows a more efficient control of the selective pressure, adjusting the population diversity as the consequence of the manipulation of the probability vector. Comparing the proposed mutation operator with that of mCGA, the new operator decreases the number of tournaments per generation and, consequently, the total number of fitness evaluations per generation. The consequence is a significant improvement in the convergence speed of the algorithm [8]. The new CGA resulting from the use of the proposed mutation is named emCGA (elitism with mutation CGA). Basically, the proposed mutation operator changes the random generation phase, by changing the chromosome just-generated with the probability vector.

3 The emCGA for the Object Recognition Problem

The problem of object recognition in images is frequently found in industrial and academic application. However, the recognition of objects using traditional search algorithms is computationally expensive. Particularly, this effort increases when it is present translation, rotation, scale variation or a partial object obstruction [6]. Therefore, the implementation of fast search algorithms for this problem is of great interest. In most cases, applications using these algorithms should be executed in real time, thus requiring fast algorithms instead of exhaustive search algorithms. Algorithms based on metaheuristics can offer reasonable performance and quality of solutions [7]. This fact motivated the use of emCGA to the object recognition problem in real images. In this work, a technique of digital processing of images is explored to extrapolate significant properties of an object in an image and to create a computational model. The proposed model, called of Light Intensity (LInt) model, is based on the intensity of the light in three channels of the image of the object to be detected, Red, Green and Blue (RGB). The RGB channels are weighted with 0.3, 0.59 and 0.11, respectively. Converting every pixel of the object image yields a generic matrix M_{illum} , where each element is a value of 256 gray levels, corresponding to the light intensity of the pixel in the same image position. When the value of the element is 0 means that the light intensity is 0%, and 255 is 100%.

The fitness function of the emCGA measures the similarity between two images. The object is defined by a reference image or, simply, reference. The input images are converted previously to the LInt model. Therefore, it is possible to reduce the computational cost and the memory demands in the fitness function execution. Based on [7], the fitness function computes the percentage of similarity between two images using the absolute error between the reference and input image models, described by equation [1]

$$S(k) = 100\% \cdot \frac{E_{max} - \{\sum_{i=1}^n \sum_{j=1}^m |F(I, k, i, j) - M_{ref}(i, j)|\}}{E_{max}} . \quad (1)$$

Where I is LInt matrix of the input image, F is function that transforms an image in a LInt model, returning its (i, j) element, M_{ref} is reference image LInt matrix, E_{max} is absolute maximum error, n are number of lines of reference image LInt matrix, m are number of columns of reference image LInt matrix and k is parameter vector of F transform.

This fitness function returns a value in the range 0..100% representing the similarity between the object and the input images. The higher this value, the higher the probability of finding the object in the input image, according to parameters vector k . Therefore, the objective is to find a vector k that yields the higher similarity measured by the fitness function.

Objects studied in this work are three-dimensional. Consequently, they can be found displaced horizontally (x axis), vertically (y axis) and in depth (z axis), as well as rotated in any of the three possible planes (angles θ , α and β). Matrix M_{ref} has n lines and m columns and all their elements are compared for computing similarity. In our experiments we considered only translations in the (x, y) plane and rotations in all planes. The F transform applies a translation and rotation operation to point (i, j) and returns the corresponding element of a input matrix I , according to equation [2] (for simplicity, this equation considers only translation and rotation in the (x, y) plane). As the values of light intensity vary from 0 to 255, the absolute maximum error of an element is 255. Therefore, the absolute maximum error for all elements (E_{max}) is defined as $E_{max} = 255 \cdot n \cdot m$.

$$F(k, i, j) = I(x', y') . \quad (2)$$

The parameters vector k is composed by the translation position (x, y) and the rotation angle θ . The position (x', y') is the transformed position of (i, j) , where $x' = x + (i \cdot \cos \theta + j \cdot \sin \theta)$ and $y' = y + (i \cdot \cos \theta - j \cdot \sin \theta)$. However, as this transformation uses trigonometrical functions that return a real number, the returned element of the matrix I is the value of the closest integer of this position, by rounding up x' and y' .

The translation position is a position in the input image and it is also the central position of the possible object location. When a position is evaluated close to the border, some positions (x', y') may be invalid. In this case the error in these points will be the maximum. This procedure allows the identification of an object partially clipped close to the borders. A chromosome in the emCGA is then defined by the parameter vector k and coded according to the following

ranges: $0 \leq x \leq c$, $0 \leq y \leq l$ and $0 \leq \theta < 360$ degrees, where c and l are the number of columns and lines of the input image. Displacements are measured in pixels (steps of 1) and angle in degrees (steps of 0.7). The overall length of the chromosome is given as $L = \log_2(c) + \log_2(l) + 9$.

The emCGA uses two parameters that need to be previously defined: the population size and the mutation rate. The search of an object in an image is a complex problem, since it is a multimodal problem, with many local maxima randomly distributed. Besides, it is important to note that the genes in the chromosome, that is, the parameters being optimized, are strongly interconnected. Consequently, the recommended mutation rate is approximately $1/L$, and the size of the population should be large enough [8].

The input images used in our experiments were 1024 x 768 bits large. In this case, parameters x and y will need 10 bits for encoding, plus 9 bits for the rotation angle. Consequently, the recommended mutation rate is approximately 3.4%. The size of the population that yielded a reasonable tradeoff between performance and computational cost was 16384 individuals. The GAs, as well as the emCGA, have their stochastic nature implemented through a pseudo-random number generator. So, it is also necessary to define a seed for this generator. For each independent run, a different random seed was chosen.

4 Experiments and Results

Three experiments were run to evaluate emCGA for the object recognition task in images. Results are presented in figures showing the reference image(s), the input image and the object (standing out in this input image using red lines). We opted to show the input in gray levels to facilitate the visualization.

Our experiments use images from digital pictures. For the sake of having a golden standard, that is, the optimal solution, for a given problem, we devised

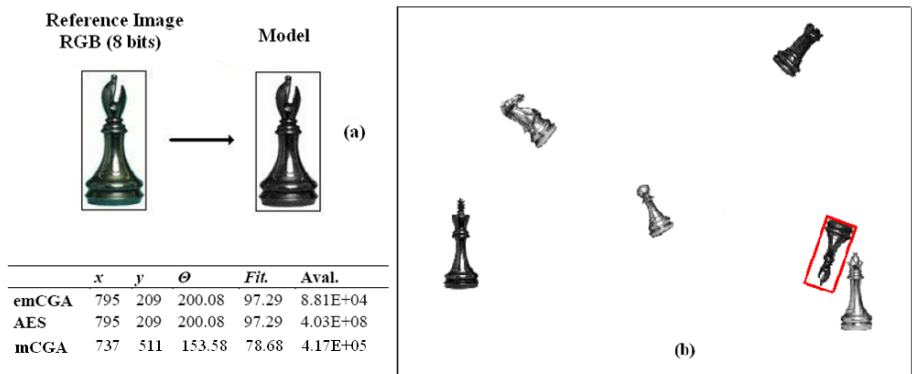


Fig. 1. Chess Bishop Search. a) Reference Image and LInt Model. b) Input Image and object recognized.

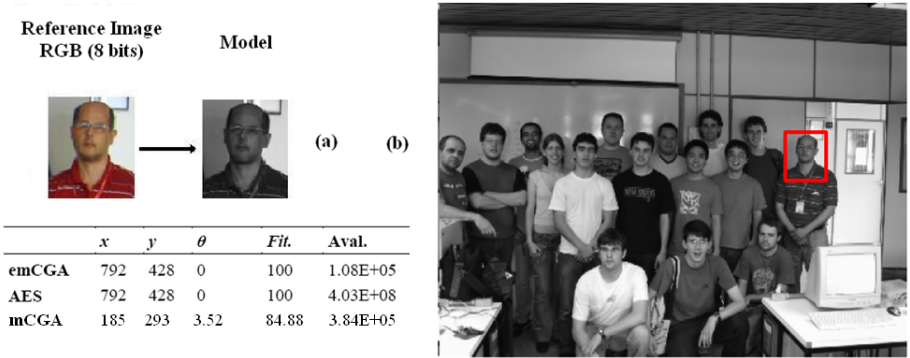


Fig. 2. Faces Search. a) Reference Image and the associated LInt model. b) Input image and the recognized face.

an Exhaustive Search Algorithm (AES). AES evaluates the same fitness function (equation 1) for all possible combinations of the parameters vector k . The results of the application of emCGA are compared with those obtained by AES and with those obtained by mCGA [3], so as to evaluate its accuracy and computational cost. Values for the parameters vectors are shown in each experiment as well as the fitness value (Fit.) and the number of fitness evaluations (Eval.).

In the first experiment, the object to be found in the input image is a chess bishop piece. The reference image is used to generate LInt model, according to figure 1a. This model is used by the emCGA to find the object in the input image. The object found is presented in figure 1b inside a rectangle to stand it out. This procedure will be also used in the remaining experiments in this work.

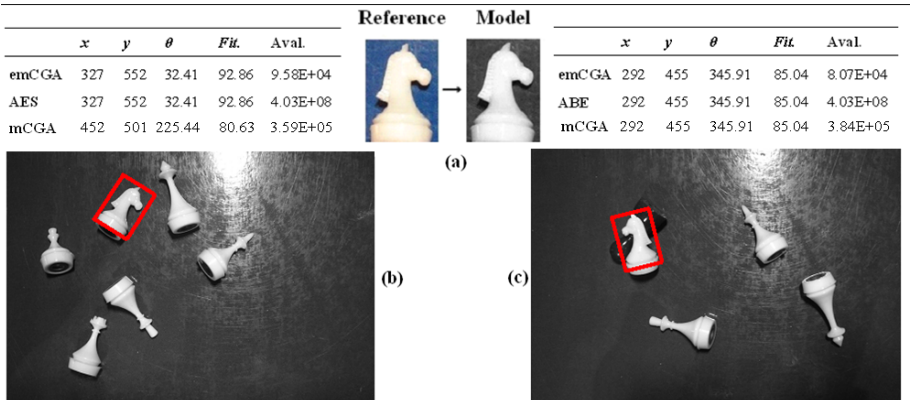


Fig. 3. Chess Knight Search. a) Reference Image and Model. b) First Input image and object recognized. c) Second Input image and object recognized.

Notice that the bishop in the input image is translated and rotated relative to the reference image.

In the second experiment, a human face is recognized in a digital picture in which there are several people. This is a difficult problem for a computer-based vision system, since many local optima can be easily realized. The face used is shown in figure 2a and input image and the recognized face are shown in figure 2b . The reference image is cut out of the input image.

In the last experiment, the object to be recognized is a chess knight piece (figure 3a). In the first input image, shown in figure 3b, the object is translated and rotated in the image plane (x, y) . The main distortion in this image that makes the problem somewhat difficult is due to a different illumination angle, shedding a bright spot to the object. In the second input image (figure 3c), the object is also rotated in both (z, y) and (x, z) axes. These rotations impose a more challenging task than the previous image. Besides these distortions, in the two input images other similar pieces are added, increasing the difficulty of the problem by injecting more local maxima.

5 Discussion and Conclusions

The object recognition problem for computational vision easily falls into an exhaustive search. It is expected that the use of a heuristic search algorithm, such as the emCGA, can achieve similar results to an AES, but with smaller computational cost. The computational costs of the algorithms are a function of the number of fitness evaluations.

The experiments were developed to evaluate the efficiency of emCGA in a real-world problem. Complex input images were used to show the robustness of the object detection method using the emCGA. In the first experiment, the chess pieces introduce several local maxima in the search space. Results shown that the emCGA reaches the global maximum with a computational cost of approximately 0.022% of the AES and 21.1% of the mCGA. However, mCGA achieved a performance around 20% worse than emCGA. In the second experiment no distortion was added, since the reference image is extracted from the input image. However, the input image has many local maxima, since there are a lot of similar faces. The emCGA was able to find the global maximum with a computational cost of approximately 0.027% of the AES. Also, emCGA was 4 times faster than mCGA, which again performed worse than emCGA. This results show that the emCGA, using the LInt model, has a good discriminatory power, capable of capturing small details in the images. The last experiment was especially devised to verify how the method behaves in the presence of significant distortions in the images. In the first input image the object to be recognized was rotated in two axes, and for the second input image, in three axes. For both cases, emCGA reached the global maximum with a computational cost of approximately 0.02% of the AES, and emCGA converged 4 to 5 times faster than mCGA. Notice that for the first input image mCGA did not found an acceptable solution for the problem, since θ is rotated around 180 degrees of the expected orientation.

Recall that, in our experiments, the chromosome of emCGA encoded only rotation in plane (x, y) . Notwithstanding, emCGA was able achieve a good result even in the presence of unexpected distortions in the 3D space.

Finally, through this work it is possible to conclude that the emCGA using the LInt model is appropriate for applications that require a compact, fast and efficient search algorithm, with limited computational resources. Therefore, we conclude that the proposed method can be efficiently applied to real-world problems without increasing significantly the implementation complexity or its computational cost.

Future work will focus in evaluating the limits of emCGA to other real problems, as well as comparing it with other similar approaches, and re-implementing the system using reconfigurable logic for real-time image processing.

References

1. Ahn, C., Ramakrishna, R.: Elitism-Based Compact Genetic Algorithms. *IEEE Trans. Evol. Comput.* 7, 367–385 (2003)
2. Ferlin, E.P., Lopes, H.S., Lima, C.R.E., Chichaczewski, E.: Reconfigurable Parallel Architecture for Genetic Algorithms: Application to the Synthesis of Digital Circuits. In: Diniz, P.C., Marques, E., Bertels, K., Fernandes, M.M., Cardoso, J.M.P. (eds.) *ARCS 2007. LNCS*, vol. 4419, pp. 326–336. Springer, Heidelberg (2007)
3. Gallagher, J., Vignaham, S., Kramer, G.: A Family of Compact Genetic Algorithms for Intrinsic Evolvable Hardware. *IEEE Trans. Evol. Comput.* 8, 111–126 (2004)
4. Harik, G., Cantu-Paz, E., Goldberg, D.E., Miller, B.L.: The Gambler’s Ruin Problem, Genetic Algorithms, and the Sizing of Populations. *Evol. Comput.* 7, 231–253 (1999)
5. Harik, G., Lobo, F., Goldberg, D.E.: The Compact Genetic Algorithm. *IEEE Trans. Evol. Comput.* 3, 287–297 (1999)
6. Jain, R., Kasturi, R., Schunck, B.G.: *Machine Vision*. McGraw-Hill, New York (1995)
7. Perlin, H.A., Lopes, H.S., Mezzadri, T.C.: Particle Swarm Optimization for Object Recognition in Computer Vision. In: Nguyen, N.T., Borzowski, L., Grzech, A., Ali, M. (eds.) *IEA/AIE 2008. LNCS (LNAI)*, vol. 5027, pp. 11–21. Springer, Heidelberg (2008)
8. Silva, R.R., Lopes, H.S., Erig Lima, C.R.: A New Mutation Operator for the Elitism-Based Compact Genetic Algorithm. In: Beliczynski, B., Dzielinski, A., Iwanowski, M., Ribeiro, B. (eds.) *ICANNGA 2007. LNCS*, vol. 4431, pp. 159–166. Springer, Heidelberg (2007)
9. Zhou, C., Meng, K., Qiu, Z.: Compact Genetic Algorithm Mutated by Bit. In: *4th World Congress on Intelligent Control and Automation*, vol. 4, pp. 1836–1839. IEEE Press, New York (2002)

A Method of Counting Pedestrians in Crowded Scenes

Byeoung-su Kim¹, Gwang-Gook Lee¹, Ja-Young Yoon²,
Jae-Jun Kim², and Whoi-Yul Kim^{1,*}

¹ Department of Electronics and Computer Engineering,
Hanyang University, Seoul, Republic of Korea, 133-791
{bskim, gglee}@vision.hanyang.ac.kr, wykim@hanyang.ac.kr

² Sustainable Architectural Engineering,
Hanyang University, Seoul, Republic of Korea, 133-791
nyx80@hanyang.ac.kr, jjkim0205@hotmail.com

Abstract. This paper proposes a method to automatically count the number of pedestrians in a video input of a crowded scene. The method proposed in this paper improves on our previous pedestrian counting method which estimates the number of pedestrians by accumulating low-level features (foreground pixels and motion vectors) on a virtual gate. To handle crowded scenes, the pedestrian counting process in this paper is weighted by the ratio of foreground pixels in the scene. The relationship between crowdedness and weighting factor is learned from 10,000 simulation images. Tests on real video sequences show that this method can successfully estimate the number of pedestrians with an accuracy of about 95%. Also, when compared to the previous method, the accuracy was increased by about 5% for highly crowded scenes. Moreover, the proposed method runs at an average rate of around 60 fps on a standard PC, which makes the algorithm realistic for multi-camera systems.

Keywords: Visual surveillance; People counting; Pedestrian flow.

1 Introduction

Counting the number of pedestrians is an important task in video surveillance because of its usefulness in applications such as marketing analysis and safety management. To date, a number of methods have been proposed to automatically estimate the pedestrian flow. Several methods have employed top-view cameras to achieve this goal [1-3]. In these methods, the number of people passing under the camera is counted by tracking segmented moving objects or blobs. Chen *et al.* [1] used an area-based method to detect the people patterns while histogram analysis was used in HSI color space to track the moving objects. Velipasalar *et al.* [2] proposed a blob tracking method to count the number of people passing through a gate. In this method, the mean shift algorithm was used to match the same objects between consecutive frames. Terada *et al.* [3] adopted a template matching technique to detect individuals where the heights of pedestrians were estimated by a triangulation method using a stereo

* Corresponding author.

camera. These top-view camera based approaches do not suffer from occlusion problems even in crowded situations because of the viewing angle. However, these systems cannot be applied to CCTV cameras (with oblique views) and therefore require installation of additional cameras.

However, other research efforts are applicable to most CCTV views [4-6]. In these methods, individuals are first detected by human models or detectors and then tracked in order to count the number of pedestrians. Masoud *et al.* [4] used difference images to detect moving blobs, and pedestrians are detected by rectangular patches based on their dynamic behavior. In addition, the Kalman filter technique has been adopted to track moving objects more accurately. Liu *et al.* [5] proposed a method based on human appearance models to detect pedestrians. In this method, human models are created using a set of low level image features, and the number of people is counted using these human models. Also, Sidla *et al.* [6] proposed a model-based method to detect the number of humans in scenes. In this method, humans were successfully detected using active shape models (ASM) even if their bodies were partially occluded. However, these model- or detector-based methods cannot be applied to multi-camera systems or embedded systems because they usually require a great amount of computational power.

Recently, we proposed a method to estimate the flow size of pedestrians based on simple pixel counting [7]. In this method, low-level features such as foreground pixels and motion vectors are used as clues to estimate the number of people passing through the gate. The number of pedestrians is estimated by accumulating the number of foreground pixels on the gate without requiring any modeling or tracking. For this reason, this method can reduce the computational complexity compared to detection or tracking based approaches. Unlike top-view camera based methods, this method can be applied to conventional CCTV systems. However, the number of pedestrians tends to be underestimated, especially in heavily crowded situations, because it does not consider the effect of occlusions between individuals.

In this paper, we propose a method to count the number of pedestrians that improves on our previous method in order to cope more effectively with crowded situations. The proposed method adopts the pixel counting process of the previous method. However, this process is further weighted by the level of crowdedness to deal with heavy occlusions. The amount of weighting is decided by the level of *crowdedness*, which is defined by the ratio of foreground pixels in the Region of Interest (ROI). With this newly introduced weighting factor, the number of pedestrians was successfully estimated, even for heavy crowded scenes. The rest of the paper is organized as follows. Section 2 reviews our previous method for pedestrian counting. Section 3 describes the proposed method for pedestrian counting which improves our previous work. The experiment results are given in Section 4, and Section 5 concludes this paper.

2 Pedestrian Counting Method Using Pixel Counting

In this section, we briefly review our previous method. The system counts the number of pedestrians who pass a pre-defined line, called a *virtual gate*. Some examples of



Fig. 1. Examples of virtual gates **Fig. 2.** Examples of feature extraction: (a) Original image. (b) Foreground image. (c) Motion vectors.

these virtual gates are shown in Fig. 1 as white lines. As shown in the figure, these virtual gates are set parallel to the dominant moving direction of the pedestrians.

Fig. 2 shows the block diagram of our previous method. The image features are first extracted and then feature extraction follows. Image features, foreground pixels and motion vectors are extracted to estimate the size of the moving objects and to distinguish the moving direction of each object. In the pedestrian counting step, the foreground pixels are accumulated along the virtual gate for continuous frames. The size of the foreground blobs is then obtained as the blobs pass the gate. Finally, the number of pedestrians is estimated using the area of the foreground regions and the expected foreground size of the humans at the gate. Also, some feature normalization processes are employed before this feature integration. To adjust to the size variation of humans caused by perspective projection, pixel sizes are normalized by their image coordinates. Different moving speeds of individuals are also considered since they produce variations in the estimated blob size.

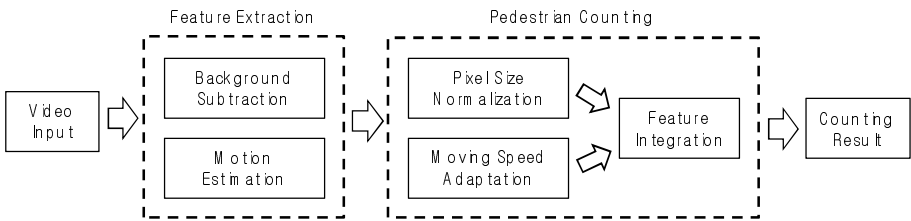


Fig. 3. Block diagram of our previous method

Details of the process are as follows. The foreground pixels are obtained using background subtraction [8]. Shadow elimination is also employed to remove shadows from the segmentation result [9]. In this method, the shadow is modeled by the assumption that the shadow region preserves the same tone of color of that of background region, but the intensity value is reduced. To determine the moving direction of each foreground pixel, optical flow is adopted [10]. Fig. 3 shows an example of the extracted features. Fig. 3(a) is the input image. Fig. 3(b) is the result of the foreground segmentation, and Fig. 3(c) is an example of the extracted motion vectors.

Once the features are extracted, the feature integration process is followed to estimate the pedestrian flow. The size of the blob passing the gate can be obtained by accumulating the foreground pixels on the gate line for continuous frames. It can be thought of as an image scanning process, but the objects are moving instead of the

scanning line. When the size of the foreground blobs is obtained, the number of people who have passed through the gate can be estimated by comparing the size of the blobs to the expected size of the humans. This process is given in Eq. (1). In this equation, $FG(x)$ is the foreground segmentation result at pixel x . The result is 1 if x is the foreground pixel but 0 if it is not. Hence, the number of pedestrians at t_{th} frame is estimated by accumulating the foreground pixel $FG(x)$ at the gate.

$$flow(i) = \sum_{x \in Gate} \alpha \cdot w_d(x) \cdot |v_x| \cdot \sin(\theta_{vx}) \cdot FG(x). \quad (1)$$

The other terms in Eq. (1) are used to fill the gap between the observations (foreground pixel counts) and the real pedestrian flow. To deal with the effect of camera projection, pixel sizes are normalized by α and $w_d(x)$ [11]. Here, $w_d(x)$ is determined as the reciprocal of the area of the elliptical human model at each pixel position. Some examples of the elliptical human models are given in Fig. 4.

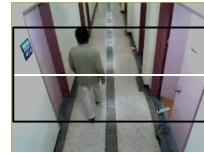


Fig. 4. An example of elliptical models for pixel size normalization **Fig. 5.** An example of ROI around the gate

As shown in Fig. 4, the sizes of the humans differ according to their positions in the image. However, this effect of perspective disappears after pixel size normalization. The constant scaling factor α was chosen experimentally to match the area of the human model to the expected size of human. Also, a motion vector is used to compensate for the effect of different moving speeds of pedestrians. As the object is moving, the pixel counting process can be considered as line sampling. If the moving speed of pedestrians is fast, it yields a smaller area of foreground blob in the pixel accumulation results, and vice versa. To compensate for the effect of different moving speeds, the magnitude of the motion vector is multiplied. The value $|V_x|$ and θ_{vx} is the magnitude and direction of the motion vector at each position. The direction of the motion vector is used to only reflect the motion component perpendicular to the virtual gate line.

3 Improvement of Pedestrian Counting Using Crowdedness

Our previous method explained in Section 2 successfully estimates the number of pedestrians when the scene is not busy. However, this method tends to underestimate the number of people in heavily crowded scenes due to the occlusions among individuals. As the number of people increases, we can easily expect that the amount of occlusion will also increase. Hence, to compensate for the pixel loss due to occlusion, the pixel counting process should be weighted when the scene is heavily crowded.

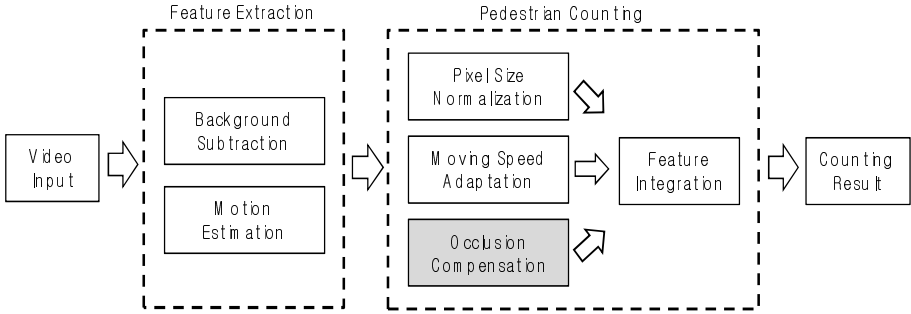


Fig. 6. Block diagram of the proposed method

Fig. 5 shows a block diagram of the proposed method. Compared to our previous method, the new contribution of this paper is in the method of occlusion compensation. This is shaded in grey in the block diagram below.

For occlusion compensation, we first defined *crowdedness* as the ratio of foreground pixels in a given Region of Interest (ROI). This ROI for crowdedness calculation is set around the gate, as shown in Fig. 6. ROI is represented as a black rectangle. This region was chosen based on the expected height of one person at the gate. Using this ROI, the level of crowdedness is computed by the ratio of foreground pixels as in Eq. (2). In this equation, x is a pixel in the ROI, and $w_d(x)$ is the weighting factor for pixel size normalization. $FG(x)$ is the foreground segmentation result for pixel x .

$$CF_i = \frac{\sum_{x \in ROI} w_d(x) \cdot FG(x)}{\sum_{x \in ROI} w_d(x)} \tag{2}$$

Unlike an uncrowded scene, it is very difficult to estimate the number of pixels in foreground in a crowded scene due to occlusions. To find the relationship between the number of pixels in the foreground and the number of people in the scene, we ran a simulation. Since we are only interested in the size of crowds in a scene, we utilized the results from the simulation to get statistically meaningful numbers. In this simulation, an imaginary camera was set at a height of 4 meters with a tilt angle of 45 degrees. In this simulation, humans are represented as ellipsoids of 1.7m in height. Using this camera setup and a human model, 10,000 artificial images were generated with a different number of people chosen randomly ranging from 1 to 60 (maximum number of people for this view). Fig. 7 shows an example image of the simulation and the result. Fig. 7(a) is an example of the simulation image, and Fig. 7(b) is the resulting graph. In Fig. 7(b), the x -axis indicates the number of people in the simulation image, and the y -axis represents the sum of the normalized foreground pixels.

As the number of people increases when crowded, the number of pixels corresponding to a human should increase but not linearly due to occlusion. As illustrated by the confidence interval in the graph in Fig. 7(b), a logarithmic function was adopted to model the relationship due to its resemblance to the shape of the graph in

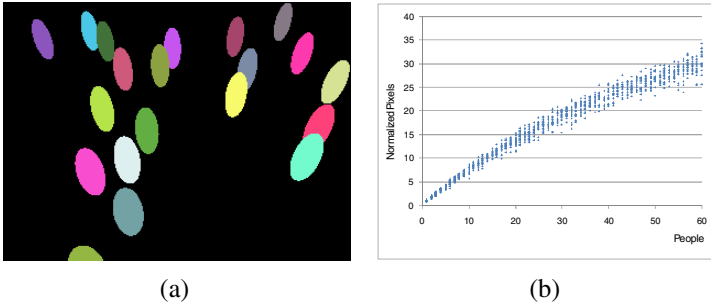


Fig. 7. The results of the simulation. (a) An example of a simulation. (b) The number of foreground pixels.

Fig. 7(b). Hence, a weighting function becomes a power form as in Eq. (3). The purpose of this weighting function is to compensate for the expected loss of the number of foreground pixels of each person due to occlusion for a crowded scene. In Eq. (3), x is the crowdedness of a scene (obtained Eq. (2)), and a , b and c are the constant coefficients of the weighting function. These coefficients are estimated by the Levenberg-Marquardt algorithm (LMA) [12].

$$w_c(x) = ax^b + c. \tag{3}$$

The LMA provides a numerical solution to find the parameters that minimize a given error function. The weighting function in Eq. (3) is trained off-line by the simulation and the same function (that is, fixed coefficients) is used for all of our experiments. In the LMA method, the error function to find the parameters in Eq. (3) is shown in Eq. (4). In this equation, $N_G(i)$ is the number of generated foreground pixels by simulation for i_{th} simulation image resulting in the number of foreground pixels without occlusions. $N_O(i)$ is the number of observed foreground pixels at the i_{th} simulation image. Fig. 8 shows the resulting weighting function obtained using LMA.

$$E = \sum_{i=1}^N \{N_G(i) - w_c(cr_i)N_O(i)\}^2. \tag{4}$$

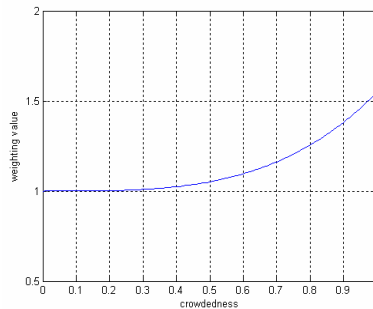


Fig. 8. The resulting weighting function to compensate for pixel loss due to occlusions

Finally, the pedestrian counting process is modified as in Eq. (5) to compensate for the miss counts due to occlusion in heavy crowds. In this equation, $w_c(cr_i)$ represents the weighting value by crowdedness cr_i . Hence, the result of the previous pixel-level feature accumulation is weighted at the frame-level by the crowdedness of the scene.

$$flow^+(i) = w_c(cr_i) \sum_{x \in Gate} \alpha \cdot w_d(x) \cdot |v_x| \cdot \sin(\theta_{vx}) \cdot FG(x). \tag{5}$$

To verify the influence of this improvement, we tested the proposed method for a simple test sequence. Fig. 9 shows examples of the test video sequence. In this video, the number of people passing the gate increases one by one from one to eight. Hence, the level of occlusion also increases as the number of people increases.

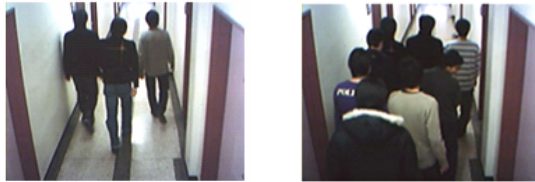
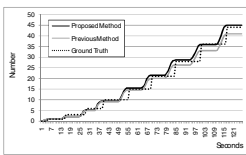


Fig. 9. An example of the test sequence

Fig. 10 shows the result of the previous method and the proposed method for the test video sequence. In Fig. 10, the results of the previous method [7] and the proposed method are represented in gray and black colors, respectively. Also, the ground truth is given as a dotted line. As shown in the figure, the previous method and the proposed method both successfully estimated the number of people who have passed when the number of people was small (where the occlusion is not heavy). However, the previous method underestimates the number of pedestrians in a crowded situation due to the occlusion among individuals. In contrast, we can see that the result of the proposed method is closer to the ground truth than our previous method, even in crowded situations.



(a) (b) (c)

Fig. 10. The results of the previous method and the proposed method
 Fig. 11. Examples of the test sequences: (a) PETS, (b) corridor1 and (c) corridor2

4 Experimental Results

To demonstrate the effectiveness of the proposed method, experiments were performed using three video sequences. Fig. 11 shows examples of each. Fig. 11(a) contains the 4th video data from the PETS2006 dataset [13]. Figs. 11(b) and 11(c) are

our own video sequences. The video in Fig. 11(b) was taken in a narrow corridor using a Logitech V-UAR38. The number of people in a scene was 8 at the maximum. The video in Fig. 11(c) was taken in a wider corridor using a Sony HDR-SR7. This video was taken just after a class finished so the number of pedestrians is quite large compared to the other sequences. The maximum number of people in one scene was 30 in this video. The resolutions of the video sequences were 320x256 for PETS and 320x240 for corridor1 and corridor2. The first 300 frames were used to build the initial background model in each video sequence.

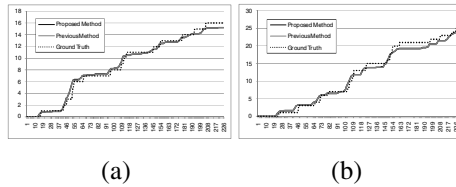


Fig. 12. PETS results

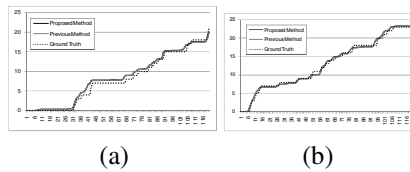


Fig. 13. Corridor1 results

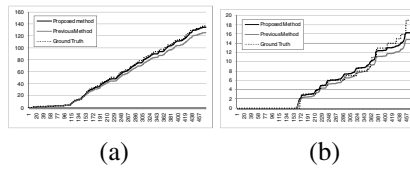


Fig. 14. Corridor2 results

To verify the performance of the proposed method, the results are compared to the ground truth. Table 1 compares the accuracy of the proposed method and the previous method. In the table, we can see that the results of the previous method and the proposed method are similar for PETS and corridor1 sequences where neither contains crowded scenes. However, the proposed method outperformed our previous method in corridor2 where the scene was extremely busy.

Figs. 12, 13 and 14 show the test results for PETS, corridor1 and corridor2 sequences, respectively. In each figure, (a) and (b) represent the number of pedestrians in different directions. That is, (a) in Figs. 12, 13 and 14 shows the number of people passing in an upward direction, while (b) shows the numbers passing in a downward direction. For video sequences using PETS and corridor1, the results of the proposed method and the previous method were almost the same because the occlusions were not severe. However, in crowded situations such as corridor2, we can see that the result of the

Table 1. Accuracy of the proposed method

<i>Video</i>	<i>Length</i>	<i>Direction</i>	<i>People</i>	<i>Previous Method</i>	<i>Proposed Method</i>
PETS	225 sec	Upward	16	15.1 (94.3%)	15.2 (95.0%)
		Downward	25	23.9 (95.6%)	24.1 (96.4%)
corridor1	119 sec	Upward	21	19.8 (94.2%)	19.9 (94.7%)
		Downward	23	23.2 (99.1%)	23.2 (99.1%)
corridor2	474 sec	Upward	137	125.5 (91.6%)	134.5 (98.1%)
		Downward	19	16.8 (88.4%)	17.9 (94.2%)

proposed method is closer to the ground truth by about five percent compared to our previous method. Notice that the great difference in Fig. 14(b) is due to the small number of pedestrians. The total number of pedestrians was 137 in Fig. 14(a) and 19 in Fig. 14(b).

For the computations, the complexity of the proposed method is quite low since only simple low-level features are used without any complex human models or detectors. Also, to reduce computation, the features are extracted only in the ROI. As a result, the proposed method runs at an average of around 60 fps on a Pentium IV 3.0GHz PC; therefore, we can expect that a single PC could manipulate the input of several cameras.

5 Conclusions

A method to count the number of pedestrians even in crowded situations is proposed in this paper. In the pedestrian counting method, the number of people passing the gate is estimated by counting the foreground pixels in a virtual gate. However, our previous method tends to underestimate the number of pedestrians, especially in heavily crowded situations, because it does not consider the effect of occlusions between different individuals. The proposed algorithm improves our previous method to deal with crowded scenes. To deal with crowded situations, we improved our method by weighting the pixel counting process by the level of crowdedness. To compensate for the influence of occlusion, a simulation was employed to learn the weighting function. Experiments on a crowded scene showed that the number of people passing through the gate was successfully estimated by using our newly introduced weighted pixel counting method. For heavily crowded video sequences, the accuracy was improved by about 5% by applying the newly introduced weighting term by crowdedness. In addition, the computational complexity is quite low in the proposed method, and it makes the algorithm realistic for multi-camera systems.

References

1. Chen, T.H., Chen, T.Y., Chen, Z.X.: An Intelligent People-Flow Counting Method for Passing Through a Gate. In: Robotics, Automation and Mechatronics 2006 IEEE Conference, pp. 1–6 (2006)
2. Velipasalar, S., Tian, Y.-L., Hampapur, A.: Automatic Counting of Interacting People by using a Single Uncalibrated Camera. In: Multimedia and Expo., 2006 IEEE International Conference, pp. 1265–1268 (2006)

3. Terada, K., Yoshida, D., Oe, S., Yamaguchi, J.: A Method of Counting the Passing People by Using the Stereo Images. *International Conference of Image Processing 2*, 338–342 (1999)
4. Masoud, O., Papanikolopoulos, N.P.: A Novel Method for Tracking and Counting Pedestrians in Real-time Using a Single Camera. *Vehicular Technology 50*, 1267–1278 (2001)
5. Liu, X., Tu, P.H., Rittscher, J., Perera, A., Krahnstoeber, N.: Detecting and Counting People in Surveillance Applications. *Advanced Video and Signal Based Surveillance*, pp. 306–311 (2005)
6. Sidla, O., Lypetsky, Y., Brandle, N., Seer, S.: Pedestrian Detection and Tracking for Counting Applications in Crowded Situations. *Advanced Video and Signal Based Surveillance*, pp. 70–75 (2006)
7. Lee, G.G., Kim, B.S., Kim, W.Y.: Automatic Estimation of Pedestrian Flow. In: *ACM/IEEE International Conference on Distributed Smart Cameras*, pp. 291–296 (2007)
8. Stauffer, C., Grimson, W.E.L.: Adaptive Background Mixture Models for Real-Time Tracking. *Computer Vision and Pattern Recognition 2*, 246–252 (1999)
9. Kim, B.S., Lee, G.G., Hong, Y.-G., Seo, H.T., Kim, W.Y.: Method of Eliminating Shadow of Moving Object for Video Surveillance. In: *IEEK Fall Conference, Korea*, pp. 768–771 (2006)
10. Lucas, B.D., Kanade, T.: An Iterative Image Registration Technique with an Application to Stereo Vision. In: *DARPA Image Understanding Workshop*, pp. 121–130 (1981)
11. Lee, G.G., Song, S.H., Kim, W.Y.: Crowd Density Estimation for Video Surveillance. In: *International Conference on Computing, Communications and Control Technologies*, pp. 196–199 (2006)
12. Levenberg, K.: A Method for the Solution of Certain Non-Linear Problems in Least Squares. *The Quarterly of Applied Mathematics 2*, 164–168 (1944)
13. PETS, Benchmark Data (2006),
<http://www.cvg.rdg.ac.uk/PETS2006/data.html>

A Novel Approximate Algorithm Used for Solving the Netting Radar Measure Elevation Angle Based on the Target Altitude*

Wen-Bo Zhao, Ji-Yan Du, Li-Ming Wang, and Yong-Ke Chen

Artillery Academy of People Liberation Army, Hefei, Anhui, 230031, P.R. China

Abstract. Solving the measured elevation angle by the netting radar from the altitude of the target is the precondition for the coordination transformation of the radar netting system. A novel approximate algorithm was proposed to be used in the paper. The simulations demonstrated that the proposed algorithm had good precision, which could be satisfied for engineering requirement of the radar netting system.

1 Introduction

The radar netting system consists of the netting fusion center, the netting radars and communication chains. The netting radars measured the target and export the corresponding plot information to the netting fusion center by the communication chains. In the netting fusion center, the plots information, firstly after transformed into the netting fusion center coordinates, then can be applied for tracking the target in the airspace by means of some data fusion algorithms. For the radar netting system, as long as transforming the plots measured by the netting radar into the inert coordinates of the netting fusion center, the targets in the airspace can be realized to be tracked. Generally with the development of modern radar technology, more and more kinds of three coordinates netting radars are added up to be composed of the data originations of the radar netting system. The height format of the plots transferred to the netting fusion center from the netting radar target is at the form of the altitude. Turning the altitude of the target into the radar measure elevation angle is the key part of the coordinate transform procedure. In this paper, a novel approximate algorithm is proposed to solve the radar elevation angle from the altitude for the given target in the airspace.

2 The Theory and Implementation of the Novel Approximation Algorithm

2.1 Introduction of the Coordinates and the Coordinate Transformation Technology

The proposed approximate algorithm is relation with the geocentric coordinates and the radar coordinates. In order to discuss the novel algorithm conveniently,

* This work was supported by the Natural Science Foundation of Anhui Province of China under grant No. 070412040.

the two kinds of coordinates and corresponding coordinate transform algorithm are introduced in this section.

(I) **The geocentric coordinates.** The origin of the geocentric coordinates is the center of mass of the earth ellipsoid. The geocentric coordinates is the coordinates built up based on the relation between the earth rotation axes and the earth equator. The geocentric coordinates can be divided into the geocentric earth coordinates and the geocentric Cartesian (inert) coordinates. For the given point of the earth, its geocentric earth coordinate is expressed as the $[L, B, H]^T$ [1]. The given point T of the earth can be expressed in the form of the geocentric inert coordinate as $[X_e, Y_e, Z_e]^T$. According to Reference [2], for the given point of the earth, transforms between the geocentric earth coordinates and the geocentric inert coordinates are determined by

$$\begin{cases} X_e = (N + H) \cos(B) \cos(L) \\ Y_e = (N + H) \cos(B) \sin(L) \\ Z_e = (N(1 - e^2) + H) \sin(B) \end{cases} \tag{1}$$

$$\begin{cases} L = \arctg\left(\frac{Y_e}{X_e}\right) \\ H = \frac{\sqrt{(X_e)^2 + (Y_e)^2}}{\cos(B)} - N \\ B = \arcsin\left(\frac{Z_e}{R}\right) + \frac{\Delta B_0}{(1 - 2k \cos(2B') + 2k^2 \sin^2(\bar{B}))} \\ \Delta B_0 = \arcsin\left[\frac{k \sin(2\bar{B})}{\sqrt{1 - e^2 \sin^2(\bar{B})}}\right] \\ \bar{B} = \arcsin\left(\frac{Z_e}{R}\right) \\ R = \sqrt{(X_e)^2 + (Y_e)^2 + (Z_e)^2} \end{cases} \tag{2}$$

where $k = 0.5e^2 \frac{a}{R}$, a is the radius of the long axes of the earth ellipsoid. e^2 is the square of the first eccentricity of the earth, $N = \frac{a}{\sqrt{1 - e^2 \sin^2(B)}}$ is the curvature radius of the prime vertical of the earth.

(II) **The radar coordinates.** The radar coordinates [3,4] can be divided into the radar inert coordinates and the radar polar coordinates. The radar inert coordinates is at the origin of the center of the radar antenna, i.e. ENU [5] coordinates. The origin of the radar polar coordinates is the same as the one of the radar inert coordinates, and the polar axes points to the north pole of the earth. The components definition of the radar polar coordinates is described as the followings. For the given target T , The range R is the distance between the radar and the target, the bearing angle A is the clockwise rotation angle between Y -axis and the radar-target -line, the elevation angle φ is the angle between the horizontal plane and the direction which the radar points at. Another coordinates introduced in this paper is the radar report coordinates. The origin, the range component and the bearing component of the radar report coordinates are the same as the ones of the radar polar coordinates. The third component of the radar report coordinates is the target altitude, different from the radar

polar coordinates. For the given point, transforming the radar polar coordinate $[R, \theta, \varphi]^T$ into the radar inert coordinate $[X_r^{(T)}, Y_r^{(T)}, Z_r^{(T)}]^T$ is determined as

$$\begin{cases} X_r^{(T)} = R \cos \varphi \sin A \\ Y_r^{(T)} = R \cos \varphi \cos A \\ Z_r^{(T)} = R \sin \varphi \end{cases} \tag{3}$$

Transforming the radar inert coordinate $[X_r^{(T)}, Y_r^{(T)}, Z_r^{(T)}]^T$ into the radar polar coordinate $[R, \theta, \varphi]^T$ is given by

$$\begin{cases} R = \sqrt{(X_r^{(T)})^2 + (Y_r^{(T)})^2 + (Z_r^{(T)})^2} \\ A = \arctg\left(\frac{X_r^{(T)}}{Y_r^{(T)}}\right) \\ \varphi = \arcsin\left(\frac{Z_r^{(T)}}{\sqrt{(X_r^{(T)})^2 + (Y_r^{(T)})^2 + (Z_r^{(T)})^2}}\right) \end{cases} \tag{4}$$

With the given target T , the radar inert coordinate marked as $[X_r^{(T)}, Y_r^{(T)}, Z_r^{(T)}]^T$, according to Reference [6], the geocentric inert coordinate $[X_e^{(T)}, Y_e^{(T)}, Z_e^{(T)}]^T$ can be expressed as

$$\begin{bmatrix} X_e^{(T)} \\ Y_e^{(T)} \\ Z_e^{(T)} \end{bmatrix} = M_{er}^T \begin{bmatrix} X_r^{(T)} \\ Y_r^{(T)} \\ Z_r^{(T)} \end{bmatrix} + \begin{bmatrix} X_e^{(r)} \\ Y_e^{(r)} \\ Z_e^{(r)} \end{bmatrix} \tag{5}$$

where $M_{er} = \begin{bmatrix} -\sin(L_r) & \cos(L_r) & 0 \\ -\sin(B_r)\cos(L_r) & -\sin(B_r)\sin(L_r)\cos(B_r) & \sin(B_r) \\ \cos(B_r)\cos(L_r) & \cos(B_r)\sin(L_r)\sin(B_r) & \cos(B_r) \end{bmatrix}$ denotes the rotation matrix from the geocentric inert coordinates to the radar inert coordinates at the place of the radar, where L_r and B_r is representatively the earth longitude and the earth latitude of the radar, where $[X_r^{(T)}, Y_r^{(T)}, Z_r^{(T)}]^T$ is the radar inert coordinate of the target T , $[X_e^{(r)}, Y_e^{(r)}, Z_e^{(r)}]^T$.

2.2 The Basic Theory of the Proposed Approximate Algorithm

As shown as Fig.1, suppose that O be the origin of the geocentric coordinates, that O' be the origin of the radar coordinates, that the altitude of the radar be h , that for a given target T the radar report coordinate be $[R, A, H]^T$, that intersection between the line from the target T to the earth centroid and the earth ellipsoid surface be the point C , the intersection between the connection line from the origin of the radar coordinates to origin of the geocentric coordinates and the earth ellipsoid surface be D , the radar measure elevation angle φ to be solved. Shown as Fig.1, in the triangle $\Delta OO'T$ if the

three borders are given in length, according to the law of cosines, the three apical angles can be calculated. In this triangle $\Delta OO'T$, it is given that $O'T = R, OO' = a_r + h, OT = a_T + H, \angle OO'T = \frac{\pi}{2} + \varphi$, if only a_r and a_T found out, the elevation angle φ will be gained. In the next part, we discuss how to gain a_r and a_T . It can be seen that both a_r and a_T are the distance from the points of the earth ellipsoid surface to the origin of the geocentric coordinates, which are called an equivalent geocentric radius in this paper. Therefore, a_r is the equivalent geocentric radius of the radar, and a_T is the equivalent geocentric radius of the target. In the following, solve procedure of a_r and a_T will be described. Suppose that the geocentric earth coordinate of the netting radar be $[L_r, B_r, h]^T$, $[L_r, B_r, 0]^T$ be the geocentric earth coordinate of the intersection D between the connecting line from the radar to the centroid of the earth and the earth ellipsoid surface. Then the geocentric inert coordinate of the point D is given by

$$\begin{cases} X_e^{(D)} = N_r \cos(B_r) \cos(L_r) \\ Y_e^{(D)} = N_r \cos(B_r) \sin(L_r) \\ Z_e^{(D)} = N_r (1 - e^2) \sin(B_r) \end{cases} \tag{6}$$

Then, according to the above definition of the equivalent geocentric radius, the radar equivalent geocentric radius a_r is given by

$$a_r = \sqrt{\left(X_e^{(D)}\right)^2 + \left(Y_e^{(D)}\right)^2 + \left(Z_e^{(D)}\right)^2} \tag{7}$$

Inserting $X_e^{(D)}, Y_e^{(D)}$ and $Z_e^{(D)}$ of (6) into (7), a_r can be written by

$$a_r = N_r \sqrt{\cos^2(B_r) + (1 - e^2)^2 \sin^2(B_r)} \tag{8}$$

where, $N_r = \frac{a}{\sqrt{1 - e^2 \sin^2(B_r)}}$ is the curvature radius of the prime vertical of the earth at the location of the netting radar. Suppose that intersection between the connecting line from the target to the earth centroid and the earth ellipsoid surface be the point C , marked as $[L_T, B_T, 0]^T$ in the geocentric earth coordinate, in the same manner of (8), the equivalent geocentric radius a_T of the target T can be expressed by

$$a_T = N_T \sqrt{\cos^2(B_T) + (1 - e^2)^2 \sin^2(B_T)} \tag{9}$$

It can be seen that the geocentric earth coordinate of the point C can be difficult to be solved accurately. Our aim is searching an approximate point C' replacing for the point C . Therefore, a projection target T' , which is the projection of the target T in XOY of the netting radar inert coordinates, is proposed in this paper. The polar coordinate of the projection target T' is $[R, A, 0]^T$, that is, the projection target T' has the same range R and bearing angle A as the target T with the difference of the radar elevation angle. Suppose that the intersection between the connecting line from the projection target to the earth centroid and the earth ellipsoid surface be C' . Generally for most of ground intelligence radars

the altitude scope of the target, which can be measured, is ranged from several kilometers to tens of kilometers. In this situation, the point C' is very close to the point C in the location, that is, C' can be substituted for C . According to (3), the radar inert coordinate of the projection target T' $\left[X_r^{(T')}, Y_r^{(T')}, Z_r^{(T')} \right]^T$ can be derived by

$$\begin{aligned} X_r^{(T')} &= R \sin(A) \\ Y_r^{(T')} &= R \cos(A) \\ Z_r^{(T')} &= 0 \end{aligned} \tag{10}$$

Using the coordinates transform algorithm from the radar inert coordinate to the geocentric earth coordinate, introduced above, the geocentric earth coordinate of the projection target T' , $[L_{T'}, B_{T'}, H_{T'}]^T$, can be solved, then the equivalent geocentric radius of the target T can be approximated by

$$a_T \approx N_{T'} \sqrt{\cos^2(B_{T'}) + (1 - e^2) \sin^2(B_{T'})} - H_{T'} \tag{11}$$

where $N_T = \frac{a}{\sqrt{1 - e^2 \sin^2(B_{T'})}}$ In the triangle $\Delta OO'T$, according to the law of cosines, the following result can be given by

$$\cos\left(\frac{\pi}{2} + \varphi\right) = \frac{(a_r + h)^2 + R^2 - (a_T + H)^2}{2R(a_r + h)} \tag{12}$$

It is known that $\cos\left(\frac{\pi}{2} + \varphi\right) = -\sin(\varphi)$, the approximate elevation angle φ is determined by

$$\varphi = \arcsin\left(\frac{(a_T + H)^2 - (a_r + h)^2 - R^2}{2R(a_r + h)}\right) \tag{13}$$

2.3 The Implementation of the Proposed Approximate Algorithm

Based on the above analysis, the detail procedure of solving the radar measure elevation angle from the target altitude can be summarized as the followings. Step 1, The computation of the radar equivalent geocentric radius a_r by means of (8). Step 2, The computation of the radar inert coordinate for the projection target T' Using (10). Step 3, The computation of the geocentric inert coordinate $\left[X_e^{(T')}, Y_e^{(T')}, Z_e^{(T')} \right]^T$ for the projection target T' according to (11), (5). Step 4, The computation of the geocentric earth coordinate for the projection target T' By means of (2). Step 5, The computation of the equivalent geocentric radius a_T of the projection target according to (9). Step 6, The computation of the radar elevation angle φ for the given target T By means of (13).

3 Simulations and Results

In this simulations, the location of the netting radar is set at point $[118^\circ, 32^\circ, 120]^T$. The grid testing network is formed with the 100 same parts in the latitude period of $[112.7^\circ, 123.3^\circ]$ and with the 100 same parts in the longitude period of $[27.5^\circ, 36.5^\circ]$. For the given target height layer, i.e., altitude 8000m, there is a rectangle testing networks in order to test the validity of the proposed approximated algorithm. The used parameters in the simulations are as followings: $a = 6378137.0m, e^2 = 0.00669437999013m$. Firstly for every testing points transform the geocentric earth coordinate into the polar coordinate, in which the corresponding elevation angles φ is used as the corresponding true elevation angle measured by the netting radar. Then for every testing point transform the radar polar coordinate into the radar report coordinate. Lastly by means of the proposed approximated algorithm in this paper, based on the radar report coordinate of every testing points, compute the radar measure elevation angle φ' , at the evaluation criterion of the absolute difference between φ' and φ , i.e., $|\varphi' - \varphi|$. For the four testing networks, testing results shown as Fig.2, some conclusions are summarized as followings.

(A) For the four testing networks, totally 101×101 test points, the elevation angle errors by the proposed approximate algorithm are controlled in the level of 10^{-5} (degree), that is, our algorithm had considerable good precision.

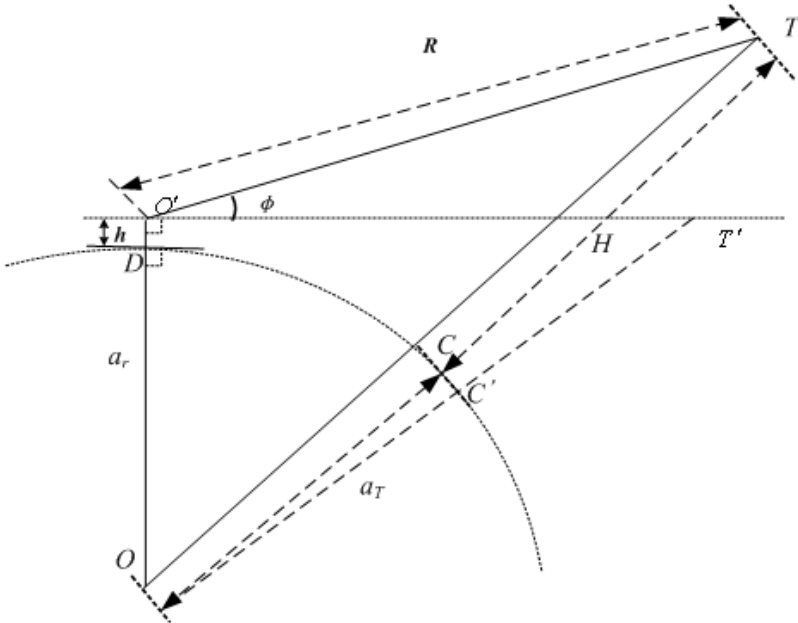


Fig. 1. For the target T , the radar measure elevation angle φ , and the altitude H

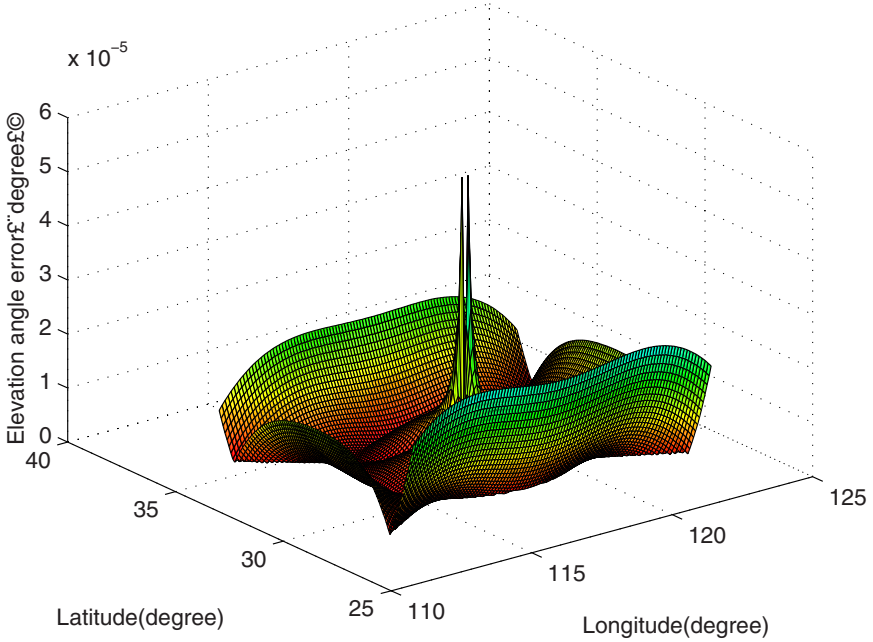


Fig. 2. The tested result of the proposed algorithm for the altitude 8000m of the target

(B) In the top blind area of the netting radar, the errors of the proposed algorithm are a little greater than the other area. In fact, the netting radar do not report the netting fusion center measured plots. Therefore, the condition does not appear in the application of the radar netting system.

(C) Apart from the top blind area of the netting radar, with the testing points distance from the netting radar added up, the errors of the proposed algorithm had some increased. However, the maximum error of the algorithm does not exceed 2×10^{-5} (degree) even though at the four fringes of the testing networks, where distance from the netting radar over 480 kilometers.

(D) Apart from (B),(C), the errors of the proposed algorithm are limited in the level of 10^{-6} (degree), that is, the proposed algorithm can satisfied the engineering requirement of the netting radar system.

References

1. Tong, X.R., et al.: GPS/INS Combination Navigation & Location and Application. Change Sha: blishing House of University of Science and Technology of National Defense (1998)
2. Zhang, S.X.: The Theory and Aplication of GPS Satellite Position. Change Sha: blishing House of University of Science and Technology of National Defense (1996)
3. Zhou, H.R., Jing, Z.L., Wang, D.P.: Tracking the moreover Target. Publish House of National Defense Industry, Beijing (1991)

4. Zhang, N.H.: The Coordinate Transformation Application in the Radar Display. *Modern Radar*. 5, 30–32 (2002)
5. He, Y., Xiu, J.J., Zhang, J.W., et al.: *Radar Data Processing and Application*. Publishing House of Electronics Industry, Beijing (January 2006)
6. Sun, Z.K., Zhou, Y.Y., He, L.X.: *Powered & Non-powered Position Technology of Single Base & Multi Bases*. Publish House of National Defense Industry, Beijing (1996)

Automated Segmentation Using a Fast Implementation of the Chan-Vese Models

Huan Xu^{1,2} and Xiao-Feng Wang^{1,2,3}

¹ Intelligent Computation Lab, Hefei Institute of Intelligent Machines, Chinese Academy of Science, P.O. Box 1130, Hefei, Anhui 230031, China

² Department of Automation, University of Science and Technology of China, Hefei, 230027, China

³ Department of Computer Science and Technology, Hefei University, Hefei 230022, China
xuhuan@mail.ustc.edu.cn, xfwang@iim.ac.cn

Abstract. In this paper, a fast implementation method of Chan-Vese models is proposed, which does not require numerical solutions of PDEs. The advantages of traditional level set methods, such as automatic handling of topological changes, are also preserved. The whole process is described as follows: First, the Otsu thresholding method is adopted to obtain the initial contours for the following level set evolution. Then, the initial curves are evolved to approach the true boundaries of objects by using the proposed fast implementation method of Chan-Vese model. Experimental results on some real and synthetic images show that our proposed approach is capable of automatically segmenting images with a low time-consumption.

Keywords: Image Segmentation; Otsu Thresholding; Active Contours; Chan-Vese Models.

1 Introduction

Image segmentation is a fundamental problem in image processing and computer vision. It is a process of partitioning the image into some non-intersecting regions such that each region is homogeneous and none of two adjacent regions are homogeneous. Various categories of segmentation techniques have been proposed in recent years, and they all face the following two challenging issues: (1) the accuracy of the segmentation; (2) the real-time acquirement of the segmentation.

Thresholding methods [1], which make decisions based on local pixel information, are effective when the intensity levels of the objects fall outside the range of levels in the background. Due to spatial structural information being ignored, they are effective for simple images that display only small amounts of structure. Regarding segmentation as an energy minimization problem, active contour model has become an effective segmentation technique. Active contour models can be broadly categorized into two kinds: parametric active contours [2] and geometric active contours [3]. The level set method is a successful improvement to active contour model, which has become increasingly popular and has been proved to be a useful tool for image segmentation

since it was proposed by Osher and Sethian [4]. Level set methods are the ones based on active contours particularly designed to handle the segmentation of deformable structure, which display interesting elastic behaviors, and can handle topological changes. Generally, a classical level set method is to consist of an implicit data representation of a hypersurface, a set of PDEs that govern how the curve moves, and the corresponding solutions for implementing this method on computers [5]. Based on the Mumford-Shad functional [6] for segmentation, Chan and Vese [7] proposed an easily handle model, or bimodal Chan-Vese model. This method can partition a given image into two regions, one representing the objects to be detected, and the other one representing the background. However, the method is time-consuming and fails to segment images with intensity inhomogeneity.

In this paper, we proposed a modified fast Chan-Vese approach to overcome above drawbacks. The whole process is described as follows: First, the thresholding method is adopted to obtain the initial contours for the following level set evolution. Then, the initial curves are evolved to approach the true boundaries of objects by using the proposed fast implementation method of Chan-Vese model. Compared with classical Chan-Vese model, our approach could automatically get much more accurate results in a fast way.

The rest of the paper is organized as follows: Section 2 introduces the level set method based on Chan-Vese (CV) model. Section 3 presents the proposed modified CV method in detail. Experiment results are demonstrated in Section 4. Finally, Section 5 concludes the paper and discusses the future work

2 Chan-Vese Models

The Chan-Vese models are curve evolution implementations of a well-posed case of the Mumford-Shah model [6]. The Mumford-Shah method is an energy-based method introduced by Mumford and Shah via an energy function in 1989. Consider a possibly noisy or blurry two-dimensional image u_0 with image domain Ω and the segmenting curve C , the general form of the Mumford-Shah function can be written as:

$$E^{MS}(u, C) = \int_{\Omega} |u_0(x, y) - u(x, y)|^2 dx dy + \mu \int_{\Omega \setminus C} |\nabla u(x, y)|^2 dx dy + \nu \cdot Length(C) \tag{1}$$

where $\mu \geq 0$ and $\nu \geq 0$ are parameters. The Length term is a constraint on the curve's length and controls its smoothness. The other two terms can divide the image into different regions while allowing for the discontinuities along the edges. The removal of any of the above three terms in Eqn.(1) will result in trivial solutions for u and C [8]. Chan and Vese[7] proposed an algorithm for decomposing the image into two regions with piecewise constant approximations. Minimizing Eqn(1) becomes the minimization of the following energy functional:

$$E^{CV}(c_1, c_2, C) = \mu \cdot Length(C) + \nu \cdot Area(inside(C)) + \lambda_1 \cdot \int_{inside(C)} |u_0(x, y) - c_1|^2 dx dy + \lambda_2 \cdot \int_{outside(C)} |u_0(x, y) - c_2|^2 dx dy \tag{2}$$

where μ, ν, λ_1 and λ_2 are positive constants, usually fixing $\lambda_1 = \lambda_2 = 1$ and $\nu = 0$. c_1 and c_2 are the averages of u_0 inside C and outside C , respectively.

To solve this minimization problem, we use the level set method [4] to represent C , i.e., C is the zero level set of a Lipschitz function $\varphi: R^2 \rightarrow R$. Then, we can replace the unknown variable C by the unknown variable φ , and the energy functional (2) can be written as:

$$\begin{aligned}
 E_\epsilon^{CV}(c_1, c_2, \varphi) = & \mu \cdot \int_{\Omega} \delta_\epsilon(\varphi(x, y)) |\nabla \varphi(x, y)| dx dy + \nu \cdot \int_{\Omega} H_\epsilon(\varphi(x, y)) dx dy + \\
 & \lambda_1 \cdot \int_{\Omega} |u_0(x, y) - c_1|^2 H_\epsilon(\varphi(x, y)) dx dy + \\
 & \lambda_2 \cdot \int_{\Omega} |u_0(x, y) - c_2|^2 (1 - H_\epsilon(\varphi(x, y))) dx dy
 \end{aligned} \tag{3}$$

where $H_\epsilon(z)$ and $\delta_\epsilon(z)$ are the regularized approximation of Heaviside function and Dirac delta function, respectively. This minimization problem is solved by taking the Euler-Lagrange equations and updating the level set function $\varphi(x, y)$ by the gradient descent method:

$$\frac{\partial \varphi}{\partial t} = \delta_\epsilon(\varphi) [\mu \operatorname{div}(\frac{\nabla \varphi}{|\nabla \varphi|}) - \nu - \lambda_1 (u_0 - c_1)^2 + \lambda_2 (u_0 - c_2)^2] = 0 \tag{4}$$

c_1 and c_2 can be updated at each iteration by:

$$c_1(\varphi) = \frac{\int_{\Omega} u_0(x, y) H(\varphi(x, y)) dx dy}{\int_{\Omega} H(\varphi(x, y)) dx dy} \quad c_2(\varphi) = \frac{\int_{\Omega} u_0(x, y) (1 - H(\varphi(x, y))) dx dy}{\int_{\Omega} (1 - H(\varphi(x, y))) dx dy} \tag{5}$$

The main advantage for this model is that it can automatically detect interior contours, where the initial curve can be placed anywhere in the image, and detect both contours with, or without gradient [7]. Since this model assumes that an image consists of statistically homogeneous regions, with intensities in each region being a constant up to a certain noise level. Therefore, the method fails to segment images with intensity inhomogeneity and is time-consuming.

3 Our Modified Fast Chan-Vese Approach

3.1 Initial Contour Using Thresholding Segmentation

Manual definition of initial contour hinders the automation of Chan-Vese model. However, in computer vision and image processing, Otsu method [9] is used to perform thresholding, or, the transformation of a gray-level image to a binary image. So we could use the method to get an initial contour for the following level set evolution.

Otsu proposed an algorithm for automatic threshold selection from a histogram of image. Let the pixels of a given image be represented in L gray levels $[1, 2, \dots, L]$. The

number of pixels at level i is denoted by n_i , and the total number of pixels by $N = n_1 + n_2 + \dots + n_i + \dots$. Then, suppose that the pixels were dichotomized into two classes C_0 and C_1 , which denote pixels with levels $[1, \dots, k]$ and $[k+1, \dots, L]$, respectively. This method is based on a discriminant criterion, which is the ration of between-class variance and total variance of gray levels. The optimal threshold of an image depends on maximizing between-class variance to maximize the separability of the resultant classes in gray levels.

3.2 Fast implementation of the Chan-Vese Model

The Chan-Vese models are usually implemented by solving PDEs, such as the level set equations [4, 10] and Poisson equations [11]. These methods are computationally intense, although they are theoretically sound. In this section, we would discuss the proposed fast implementation method for Chan-Vese models.

Assuming that the evolving curve C in Ω is the boundary of an open set of Ω . $inside(C)$ denotes the region C and $outside(C)$ denotes the region $\Omega \setminus C$, φ is the level set function. c_1 and c_2 are the averages of u_0 inside C and outside C , respectively. They could obtain from Eqn.(5).

Furthermore, let's define:

$$c_{\max} = \max(c_1(\varphi), c_2(\varphi)) \quad c_{\min} = \min(c_1(\varphi), c_2(\varphi))$$

Then we give our evolution equation:

$$\frac{\partial \varphi}{\partial t} = (u_0 - \frac{c_{\max} + c_{\min}}{2})(c_{\max} - c_{\min}) \tag{6}$$

For numerical implementation, the Heaviside function $H(\varphi)$ here is regularized as:

$$H_\epsilon(z) = \frac{1}{2} (1 + \frac{2}{\pi} \arctan(\frac{z}{\epsilon}))$$

Our approach can easily implemented use the difference scheme and iterations:

$$\frac{\varphi_{i,j}^{k+1} - \varphi_{i,j}^k}{\tau} = (u_0 - \frac{c_{\max} + c_{\min}}{2})(c_{\max} - c_{\min})$$

$$\varphi_{i,j}^{k+1} = \varphi_{i,j}^k + \tau(u_0 - \frac{c_{\max} + c_{\min}}{2})(c_{\max} - c_{\min}) \tag{7}$$

where τ is the time step.

Here the whole process of our fast Chan-Vese method is described in Fig.1.

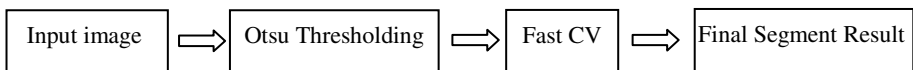


Fig. 1. The process of our method

4 Experimental Results

The proposed method is implemented on a computer which has two Intel(R) Pentium(R) 2.19GHz CPUs, 2G bytes RAM, and runs the Microsoft Windows operating system. The CPU times given in this paper are the sums of system CPU times and user CPU times. The system CPU time is usually very small, typically 0.01-0.08 seconds. In the following experiment, the parameters for all the images are $\tau = 0.1$, $\varepsilon = 1$.

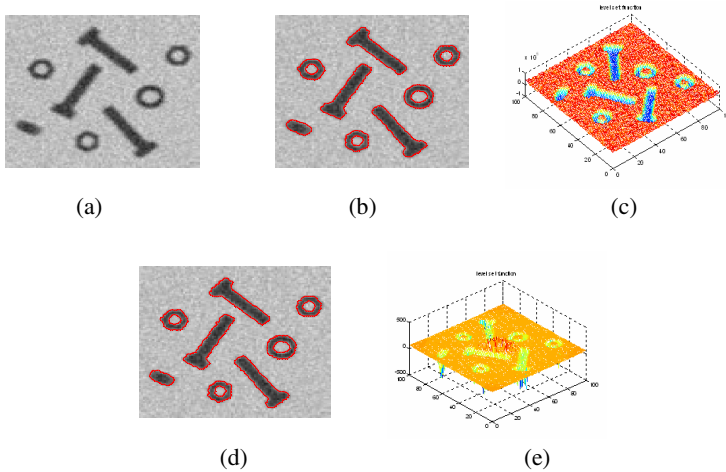


Fig. 2. Comparison of the proposed fast CV approach to the classical Chan-Vese method. (a) Original image (image size is 100x100) (b) Segmentation result using Fast CV (Elapsed time is 0.299989 seconds) (c) The level set function of fast CV (d) Segmentation result using Chan-Vese model (Elapsed time is 2.058161 seconds) (e) The level set function of Chan-Vese model.

The test images are segmented using the proposed method and the classical method with PDEs solution. Fig.2 shows that the proposed method is 10 times faster than the classical method with PDEs solution and achieves the same segmentation results, Fig.2 (b)-(c) demonstrate that the proposed method can also efficiently handle the topological changes.

The effects of intensity inhomogeneity on the proposed method are illustrated in Fig.3. The classical Chan-Vese model fails to segment the image in Fig.3(a) (as shown in Fig.3(d)). The reason is that the motion of the contours in the Chan-Vese model is guided by global image information only, which can lead to wrong segmentation result for the images with intensity inhomogeneity. In contrast, Fig.3 (b) and (c) show that the proposed fast CV method works on the image in Fig.3 (a).

We also test our approach on real image (as shown in fig.4 (a)). It can be seen that our approach can get a clear boundary with elapsed time being 0.319217 seconds, while the Chan-Vese model needs 3.585632 seconds.

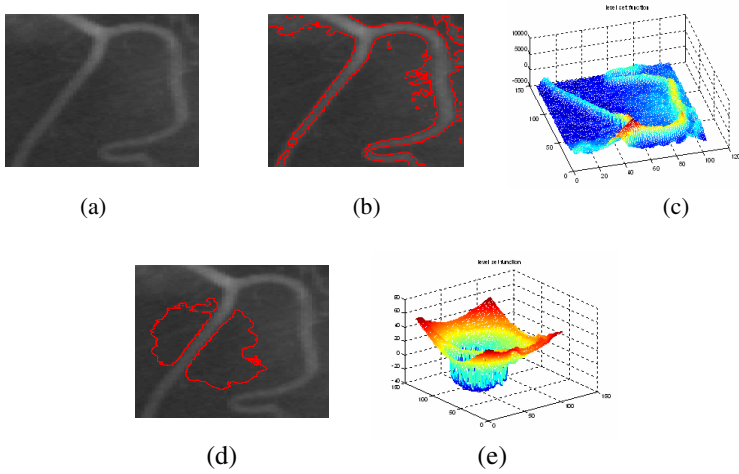


Fig. 3. Comparison of the proposed fast CV approach to the classical Chan-Vese method. (a) Original image (image size is 131x103) (b) Segmentation result using fast CV result (Elapsed time is 0.319217 seconds) (c) The level set function of fast CV (d) Segmentation result using Chan-Vese model (Elapsed time is 13.367303 seconds) (e) The level set function of Chan-Vese model.

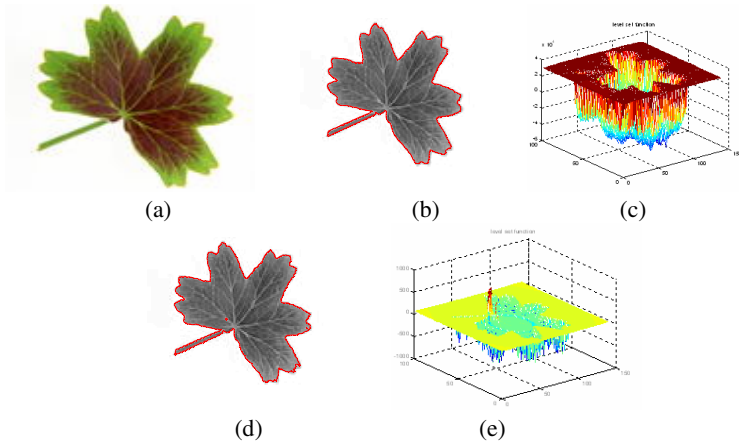


Fig. 4. Comparison of the proposed fast CV approach to the classical Chan-Vese method. (a) Original image (image size is 99x140) (b) Segmentation result using fast CV result (Elapsed time is 0.181292 seconds) (c) The level set function of fast CV (d) Segmentation result using Chan-Vese model (Elapsed time is 3.585632 seconds) (e) The level set function of Chan-Vese model.

5 Conclusion

In this paper, a fast implementation method for the Chan-Vese models is proposed, which does not require solutions of the PDEs. Otsu thresholding method is first used

to obtain the initial contour for the following level set evolution, and then the proposed modified fast Chan-Vese approach is used to evolve the initial contour with an automatic segmentation solution scheme. We have applied our approach on some synthetic and real images, experimental results show that our approach is fast and can achieve good results. Our future works will focus on extending the fast scheme to multiphase Chan-Vese model to get more satisfactory results on multiphase images.

Acknowledgements. This work was supported by the grants of the National Science Foundation of China, Nos. 60705007 & 60772130, the grant of the **Graduate Students' Scientific Innovative Project Foundation of CAS (Xiao-Feng Wang)**, the grant of the Scientific Research Foundation of Education Department of Anhui Province, No. KJ2007B233, the grant of the Young Teachers' Scientific Research Foundation of Education Department of Anhui Province, No. 2007JQ1152.

References

1. Lim, Y.W., Lee, S.U.: On the Color Image Segmentation Algorithm based on the Thresholding and the Fuzzy C-Means Techniques. *Pattern Recognition* 23(9), 935–952 (1990)
2. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. *Int. J. Comput. Vis.* 1, 321–331 (1987)
3. Caselles, V., Kimmel, R., Sapiro, G.: Geodesic Active Contours. In: *Proc. 5th International conf. on Computer Vision, Boston*, pp. 694–699 (1995)
4. Osher, S., Sethian, J.A.: Fronts Propagating with Curvature-Dependent Speed: Algorithms based on Hamilton–Jacobi Formulation. *Journal of Computational Physics* 79, 12–49 (1988)
5. Tsai, Y.H.S., Osher, S.: Total Variation and Level Set Based Methods in Image Science. *Acta Numerica*, 1–61. (2005)
6. Mumford, D., Shah, J.: Optimal Approximation by Piecewise Smooth Functions and Associated Variational Problems. *Commun. Pure Appl. Math.* 42, 577–685 (1989)
7. Chan, T.F., Vese, L.A.: Active Contours without Edges. *IEEE Trans. Image Processing* 10(2), 266–277 (2001)
8. Tsai, A., Yezzi, A., Willsky, A.S.: Curve Evolution Implementation of the Mumford–Shah Functional for Image Segmentation, Denoising, Interpolation, and Magnification. *IEEE Trans. Image Process* 10(8), 1169–1186 (2001)
9. Otsu, N.: A Threshold Selection Method from Gray Level Histogram. *IEEE Transactions on System, Man and Cybernetics* 8, 62–66 (1978)
10. Sethian, J.: *Level Set Methods and Fast Marching Methods*. Cambridge Monograph on Applied and Computational Mathematics. Cambridge University Press, Cambridge (1999)
11. Vese, L., Chan, T.: A Multiphase Level Set Framework for Image Segmentation using the Mumford and Shah Model. *Inter. J. Computer Vision* 50(3), 271–293 (2002)

Automatic Facial Feature Points Detection

Vitoantonio Bevilacqua^{1,2}, Alessandro Ciccimarra¹, Ilenia Leone¹,
and Giuseppe Mastronardi^{1,2}

¹ Dipartimento di Elettrotecnica ed Elettronica Politecnico di Bari
Via Orabona, 4 70125 – Bari, Italy
bevilacqua@poliba.it

² e.B.I.S. s.r.l. (electronic Business in Security), Spin-Off of Polytechnic of Bari,
Str. Prov. per Casamassima Km. 3-70010 Valenzano (BA), Italy

Abstract. This paper presents an algorithm which detects automatically the feature points in a face image. This is a fundamental task in many applications, in particular in an automatic face recognition system. Starting from a frontal face image with a plain background we have effected an image segmentation to detect the different facial components (eyebrow, eyes, nose, mouth and chin). After this we have searched for the feature points of each face component. The algorithm has been tested on 320 face images taken from the Stirling University Face Database [10]. The points extracted in this way have been used in a face recognition algorithm based on the Hough transform.

Keywords: facial features points; face recognition.

1 Introduction

In this paper we present an algorithm which detects automatically the feature points in a frontal face image. This is a very simple task for a human being who has no difficulty in localizing, for example, the eyes in a face. On the contrary, this is a very difficult task for a computer which can only do arithmetic and logic operations.

These feature points are important in many applications, mainly in an automatic face recognition system. Nowadays the increasing need for security makes these systems very useful. If we think of the 11th September 2001 and of all the other terrorist attacks which we hear every day, we can easily understand how important it is to identify automatically, for example, the presence of a terrorist in an airport or in a station.

The proposed algorithm can be divided into two parts. First of all we have a face segmentation in which we localize the various face components (eyebrows, eyes, mouth, nose and chin). After this, in each component, we detect 18 features points: the two pupils, the four eye corners, the four eyebrow corners, the two nostrils, the nose tip, the two mouth corners, the upper and lower lip extremity and the tip of chin.

We present the experimental results of the feature points detection in 320 images taken from the Stirling University Face Database [10].

The usefulness of the detected points is tested using a face recognition algorithm based on the Hough transform.

2 Face Segmentation

In a previous work Bevilacqua et al. showed that those five features are used in different kinds of algorithms used in face identification and addressed that problem in a face recognition system based on Pseudo 2D HMM applied to neural network coefficients [20].

The first stage of the proposed algorithm is the face segmentation, that is to say the search for the eyebrows, the eyes, the mouth, the nose and the chin. The input to this step is the face region extracted by the Kienzle, Bakir, Franz e Scholkopf library [9].

2.1 Eyes Detection

The most important step of the face segmentation is the eye band detection. To do this, the image is firstly divided into three clusters using the competitive learning algorithm developed by Mavrincac [4]. The darkest cluster contains face features such as eyes, mouth, nostrils and eyebrows. We create a binary image containing the darkest pixels. This image is eroded using a 3x3 square as a structuring element. We find the regions that probably contain the eyes by means of a template matching with an eye template. The similarity between the template and the different regions of the image is calculated using the Normalized Cross-Correlation [5]. In order to verify the real presence of the eyes in the located region, we use a Support Vector Machine classifier. The training set has 400 images, 200 containing eyes and 200 containing other parts of the face. The choice of the images which don't contain eyes is very important to improve the SVM performance.

We have first put in the training set images containing facial components which look like eyes, such as the eyebrows. The other training images have been selected following the bootstrap method described in [13] which gives the possibility to add useful examples in an incremental way.

The SVM model contains 372 support vectors. The gamma parameter of the radial basis function is 0.03125. Then we try to localize the two eyes in a more precise way: we create an eye map with the pixels belonging to regions darker than the surrounding ones [12]. Subsequently we detect the connected components matching the expected eye dimensions.

2.2 Localization of Other Face Components

Once detected the eyes, we proceed to the localization of the other facial components using simple anthropometrical considerations.

The eyebrows are detected as the two regions above the two eyes already localized. We look for the mouth in a region below the eyes. We compute the derivative of the Variance Projection Function [11] of the image. The mouth ordinate is the one which

maximizes the VPF derivative. Finally the nose is localized in a region between the eyes and the mouth.

3 Feature Points Detection

Once the face has been detected, we try to localize the different feature points. In particular we look for the two pupils, the eye corners, the eyebrow corners, the nostrils, the nose tip, the mouth corners, the upper and lower lip extremity.

3.1 Pupil

In the grey scale eye image, the iris is a circular dark region. To remove the reflex we process the image with a 5x5 minimum filter and equalize the image to improve the contrast between the iris and the rest of the image. We estimate a rough position of the pupil and the iris radius length working with binary eye images. We erode the image to increase the separation between the connected regions. We proceed labelling the connected components: we calculate the height and the width of the biggest connected component and its centre coordinates. The iris dimensions are calculated on the base of the dimension of this connected component. The pupil is localized more precisely in this way: we look for all circles having radius equal to the estimated iris radius and we find the one having the lowest average pixel value. The pupil position is computed as the average point between the previously located connected component and the centre of the darkest circle. In Fig. 1 there are some results.



Fig. 1. Examples of detected pupils

3.2 Eye Corners

Once we have located the pupil and estimated the iris radius length, we reduce the region where we can search for the eye corners considering a rectangle region centered in the pupil and we look for the eye corners in the first and last 30% of it. The researched point will have many white pixels belonging to the sclera on one side, and darker pixels belonging to the skin on the other side. So its variance will be higher.



Fig. 2. Examples of detected eye corners

We compute the average position of the 20% of the pixels with higher variance. This procedure is repeated for all the four eye corners. Fig. 2 shows some results.

3.3 Eyebrow Corners

The eyebrow is a dark region surrounded by a lighter one (the skin).

We look for the darkest regions and we create a binary image. We proceed analysing the connected components and the corners are located as the extremities of the biggest connected component.

Fig.3 shows some examples of detected eyebrow corners.



Fig. 3. Examples of detected eyebrow corners

3.4 Mouth Feature Points

The first step for the mouth feature points detection is the search for the lip cut. This is the darkest line in the mouth image.

We compute the Integral Projection Function of the mouth image and the lip cut is localized as the ordinate which minimizes the IPF. After that, we cluster the lip cut image and look for the lateral extremity of the darkest region. Then we look for the upper lip extremity: we examine the region above the lip cut, we cluster it searching the feature point in the darkest connected region. In order to find the lower lip extremity we examine the region below the lip cut: we equalize its histogram and we find the extremity of the darkest connected component again.

Fig. 4 shows these mouth feature points.



Fig. 4. Examples of detected mouth feature points

3.5 Nostrils and Nose Tip

Analysing the nose images it's clear that nostrils are two dark regions surrounded by the skin which is lighter. We find the darkest regions and the nostril are located studying the connectivity of the resulting binary image.

The nose tip will be near the nostrils, in particular above them. We examine the region above the nostrils looking for the lightest pixel.

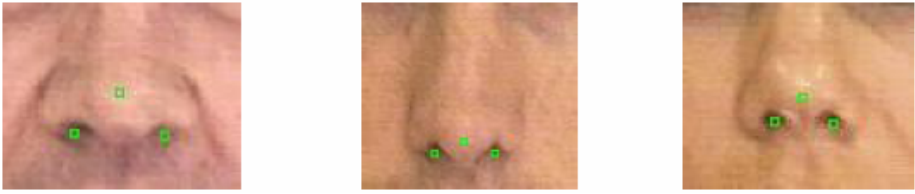


Fig. 5. Examples of detected nose tips and nostrils

In Fig. 5 there are some examples of detected nostrils and nose tips. It is clear that we cannot locate the nose tip very precisely even manually, owing to the low resolution of the examined images.

3.6 Chin

The tip of chin is identified by a deterministic algorithm based on edge analysis. It can be divided into two steps. We pre-process the chin image and then we look for the chin curve. In the first step, we smooth the chin image and we increase its luminance to highlight the chin curve. Then we detect edges using the Canny edge detector [17]. In the second step we look for a curve with an upwards concavity and an axis of vertical symmetry and located in the central search area. These steps are repeated until the chin curve, and so the tip, is identified. There are cases in which the chin curve identification fails, for example when there is a long beard or in very dark images. In these cases, we identify a hypothetical chin tip on the base on pre-inserted statistic data.

Fig. 6 shows two examples of detected chin tips. It's clear that we cannot locate the chin tip very precisely even manually, due to the low resolution of the examined images and to the presence of the shadow under the chin curve.

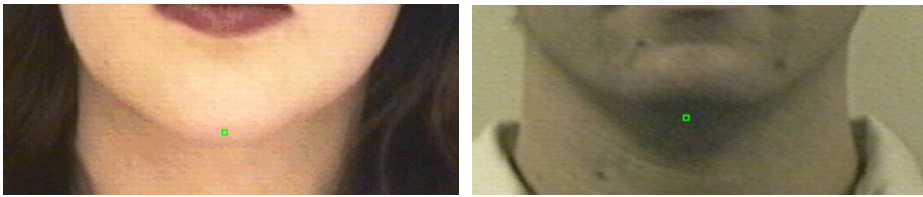


Fig. 6. Examples of detected chin tips

4 Experimental Results

The automatic feature points detection algorithm has been tested on 320 images taken from the Stirling University Face Database available on-line [10].

The errors have been calculated as the distance in pixel between the manually located points and the ones automatically obtained with the developed algorithm. We have also tested the robustness of the algorithm with noisy images (2% salt and pepper noise).

Table 1 presents the errors found and Fig. 7 shows some faces with all the detected points.

Table 1. Errors

Feature point	No noise (px)	Salt and pepper noise 2% (px)
Right eye pupil	2.07	2.67
Left eye pupil	2.60	2.86
Right eye outer corner	3.52	3.76
Right eye inner corner	4.07	5.14
Left eye outer corner	4.58	6.63
Left eye inner corner	4.14	6.14
Right eyebrow outer corner	13.02	14.57
Right eyebrow inner corner	12.08	13.85
Left eyebrow outer corner	10.29	11.93
Left eyebrow inner corner	14.75	13.73
Left nostril	5.50	13.86
Right nostril	4.67	12.53
Nose tip	6.72	13.46
Left mouth corner	4.31	4.67
Right mouth corner	4.39	5.15
Top mouth	5.90	6.12
Bottom mouth	5.45	5.96
Tip of chin	7.23	10.15

The feature points detected in this paper have been used in a face recognition algorithm based on the Hough transform. This algorithm is similar to the classical GHT one. The difference is that the reference table is indexed with the theta angle that the vector of the model point in exam forms with the horizontal axis [14]. We obtained a correct matching in the 80% of the examined cases.

**Fig. 7.** Examples of detected feature points



Fig. 7. (continued)

5 Conclusions and Future Works

In this work we have developed an algorithm to automatically locate the face feature points. This is a very simple task for a human being, but it is extremely complicated for a computer.

We have firstly segmented the face to locate the different components (eyes, eyebrows, nose, mouth) and then we have detected the feature points.

From the results we can see that the most difficult points to localize are the eyebrows corners: this is due to the low resolution of the examined images which does not give the possibility to precisely estimate the beginning and the end of the eyebrow.

A possible future development of this work is the generalization to images with unconstrained background and with partial occlusion like beard and spectacles. We are also going to improve the chin detection.

References

1. Zhao, W., Chellappa, R., Rosenfeld, A., Phillips, P.J.: Face Recognition: A Literature Survey. *ACM Computing Surveys* (2003)
2. Heisele, B., Ho, P., Poggio, T.: Face Recognition with Support Vector Machines: Global versus Component-based Approach. In: *Proceedings of IEEE International Conference on Computer Vision (ICCV 2001)* (2001)
3. Lanzarotti, R.: Facial Features Detection and Description, Master thesis, Università degli studi di Milano (2003)

4. Mavrincac, A.: Competitive Learning Techniques for Color Image Segmentation. *Machine Learning and Computer Vision* (2007)
5. Lewis, J.P.: Fast Normalized Cross Correlation. *Industrial Light and Magic* (1995)
6. Hsu, C., Chang, C., Lin, C.: A Practical Guide to Support Vector Classification (2007)
7. Feng, G.C., Yuen, P.C.: Variance Projection Function and Its Application to Eye Detection for Human Face Recognition. *Pattern Recognition Letters* (1998)
8. Vezhnevets, V., Degtiareva, A.: Robust and Accurate Eye Contour Extraction. In: *Proc. Graphicon* (2003)
9. Kienzle, W., Bakir, G., Franz, M., Scholkopf, B.: Face Detection - Efficient and Rank Deficient. *Advances in Neural Information Processing Systems* (2005)
10. Stirling University Face Database, <http://pics.psych.stir.ac.uk>
11. Feng, G.C., Yuen, P.C.: Variance Projection Function and Its Application to Eye Detection for Human Face Recognition. *Pattern Recognition Letters* (1998)
12. Wu, J., Zhou, Z.: Efficient Face Candidates Selector for Face Detection (2002)
13. Sung, K.K., Poggio, T.: Example-based Learning for View-based Human Face Detection. *IEEE Trans. Pattern Analysis and Machine Intelligence* (1998)
14. Mastronardi, G., Daleno, D., Bevilacqua, V., Chiaia, G.: Tecniche di identificazione personale basate sulla trasformata generalizzata di Hough applicata a nuvole di punti-Congresso Nazionale. AICA 2007 - Cittadinanza e Democrazia Digitale, Milano, September 20–21 (2007)
15. Wiskott, L., Fellous, J., Kruger, N., Malsburg, C.: Face Recognition by Elastic Bunch Graph Matching. *Intelligent Biometric Techniques in Fingerprint and Face Recognition* (1999)
16. Mian, A., Bennamoun, M., Owens, R.: An Efficient Multi-modal 2D-3D Hybrid Approach to Automatic Face Recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence* (2007)
17. Canny, J.: A Computational Approach to Edge Detection. *IEEE Trans. PAMI* 8(6) (1986)
18. Ciccimarra, A.: Ricerca Automatica dei punti caratteristici del volto. First Level Degree Thesis (2008)
19. Delac, K., Grgic, M.: A Survey of Biometric Recognition Systems (2004)
20. Bevilacqua, V., Cariello, L., Carro, G., Daleno, D., Mastronardi, G.: A Face Recognition System Based on Pseudo 2D HMM Applied to Neural Network Coefficients. *Soft Comput.* 12(7), 615–621 (2008)

Automatic Subspace Synthesis of Motion Styles Based on Isomap*

Jian Xiang¹ and Hongli Zhu²

¹ School of Information and Electronic Engineering, ZheJiang University of Science and Technology, 310023, HangZhou, China
freexiang@gmail.com

² College of Information & Electrical Engineering, ZheJiang University City College, 310015, HangZhou, China

Abstract. Motion synthesis methods based on 3D motion data are extensively studied in recent years. And 3D motion styles synthesis has been rarely explored. In this paper, a method for automatic and quantitative synthesis human motion styles is proposed. First extended ISOMAP theory is used to map original motions into low-dimensionality subspaces, which can reserve the intrinsic properties of original data. Synthesis is applied in such new subspaces and motion styles can be reconstructed. Experimental results show that our methods are effective.

Keywords: Motion style; subspace; synthesis; Isomap.

1 Introduction

In recent years, synthesis motion styles synthesis are very important for computer animation production systems. A large number of characters with different motion styles are needed for animation research and production. So motions with desired styles by synthesizing existed motion from 3D motion databases are needed.

But for some motions with exaggerated styles which can not be performed by the actor, we need to synthesis original data for motion styles. In order to do these, following challenges should be focused on: first motion data is high-dimensional data, For example, A motion with 151 frames and 51 DOFs is represented as a vector with 7701 dimensions. Since semantics of motion data is not obvious, relationship between original motion data is complex. Expecially, 3D motion styles are very abstract and higher-level semantics, so it's even much more difficult to uncover the intrinsic structure.

Since the dimension of motion feature extracted is very high, the distances between each two motion data are almost the same and cannot be discriminated, which is called "Curse of dimensionality (Beyer et al., 1999)", and high dimensional data also result in more computational complexity. Therefore some dimensionality reduction

* This work is supported by Research Planning Project of Department of Education of ZheJiang Province(20070777).

techniques are necessary to uncover intrinsic properties of original data in low-dimensionality subspace. Principle Component Analysis (PCA) is such a method. With PCA a subspace can be constructed, which can be used to describe original data space approximately. In general, the dimension of PCA subspace is much smaller than that of original data space.

Since human motion data have non-linear intrinsic structures that are invisible to linear subspace, here we use extended non-linear Isomap to map original motion clips into low-dimensional subspace.

Motion style is an abstract concept, so quantitative analysis of motion style is different from that of low-level motion data. So we decompose the style and content parameters in the nonlinear subspace which map between a learned unified nonlinear low-dimensionality subspace and the visual input space. According to adjust these parameters, new motion styles can be reconstructed.

With motion capture systems widely used to acquire realistic motion data, realistic and highly detailed motion segments are commercially available and widely used to produce animations characters in many applications, such as simulations and animation films. And researchers have focused their researchs on manipulating 3D Motion data in recent years.

[1] proposed multi-resolution signal processing method to motion data. signal blending and reshaping methods are applied on motion data to get new results. A motion transition method[2] is given by which efficient and smooth transitions between different motion clips can be achieved. [3] introduced a data structure, which includes all information should perform blending operations on motions: In [4] a novel method, motion graphs, is introduced for creating realistic, controllable motion. [5] presented a motion style interpolation method. They use RBF to do style interpolation. [6] also presented a style-based motion synthesise framework, where SGPLVM is used as the probability model for pose selection when solving IK.

[7] provided an automated method for identifying logically similar motions in a data set and using them to build a continuously and intuitively parameterized space of motions, which allows users to control motions parametrically. But in this method motion style was not considered. Above methods focus on editing motion capture data, interpolation methods are applied in them. These methods are difficult to set some style parameters. [8] were learning motion patterns from a highly varied set of motion capture sequences.

Since human motion data have non-linear intrinsic structures that are invisible to linear subspace, non-linear subspace are more suitable for 3D human motion synthesis than linear subspace(like PCA). In this paper, synthesis method for new motion styles based on extended Isomap subspace is proposed.

2 Motion Representation and Feature Extraction

In this paper, By motion capture system, each motion is presented by the skeleton with 51 DOFs (corresponding to 16 joints of human body). So in original motion data space, each frame of motion clip is represented as a vector with 48 dimensions.

Here we extract additional features of the skeletal motion by 16 joints of original motion data, they include: (1) joint positions, (2) joint angles, (3) joint velocities. These three aspects were found to be sufficient in experiments.

Joint velocity is approximated using the position differences between the pose before and after the

Given frame, in total, we have 76 features. A motion clip with n frames is represented as a vector with $n \times 76$ dimensions. The data in such high-dimensional space has much more computation time and complex structure which is difficult to analyze. So dimensionality reduction techniques which map original data into low-dimensionality subspace should be introduced in the next section.

3 Isomap Subspace Generated

3.1 Isomap

We use Isomap to generate new subspace, which extends MDS by sophisticated distance measurement to achieve nonlinear mappings. A data graph is built that is locally connected within the neighborhoods of each point, and then the pairwise distances are measured by the length of the shortest path on that graph which is an approximation of the distance between its end points, as measured within the underlying manifold. Finally, classical MDS is used to find low-dimensional points with similar pairwise distances. Isomap can not only reduce the dimensionality of high-dimensional input space, but also find meaningful low-dimensional structure hidden behind these original observations.

After non-linear reduction by Isomap (Fig.1), the low-dimensional embedding of the original motion clip is obtained with a very simple structure. Experiments showed that when 72 motion features are embedded to 7 or 8 low-dimensional space, residual error is minimal.

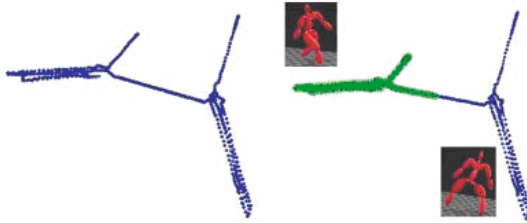


Fig. 1. Embedding to three dimension subspace by Isomap

But geo-distance of Isomap is only defined on training sets and raw Isomap cannot map new samples to the embedding space because it requires a whole set of points in the database to calculate geo-distance. So new queries outside the given database cannot be handled by raw Isomap. So classical Isomap need to extended to non-training set, which is described in next section.

3.2 Generalization of Isomap

In the above section, every motion in database is mapped into the new low-dimensionality subspace. But for a new motion outside the motion database, it is

unclear how to evaluate its map in the Isomap embedding subspace. Here we applies neural network to approximate the optimal mapping function. The optimal mapping function f

$$f = \arg \min_f \sum_{i=1}^m \left\| f(x_i) - z_i \right\|^2$$

Where m is the number of motions in the database. Clearly, this is a multivariate non-parametric regression problem, since there is no a priori knowledge about the form of the true mapping function which is being estimated.

In this work, we use radial basis function(RBF) networks, and the standard gradient descent is used as a search technique. The mapping function learned by RBF networks can be represented by

$$f_i(x) = \sum_{j=1}^h \omega_{ij} G_j(x)$$

Where h is the number of hidden layer neurons, $\omega^{ij} \in \mathbb{R}$ are the weights. G^i is the radial function defined as follows:

$$G_i(x) = \exp\left(-\frac{\left\| x - c_i \right\|^2}{\sigma_i^2}\right)$$

Where c^i is the center for G^i , and σ^i is the basis function width. The k -dimensional mapping in the embedding subspace can be represented as follows:

$$x \rightarrow f(x) = (f_1(x), f_2(x), \dots, f_k(x))$$

Where $f = [f_1, f_2, \dots, f_k]$ is the mapping function. Since the mapping function is approximated by the RBFNN(radial basis function neural network), we call the new embedding subspace RBFNN space.

In above mentioned, the RBFNN approximates the optimal mapping function from the original motion space to RBFNN low-dimensionality subspace. It is trained with the training sample between original motion space and Isomap mapping subspace. The motion representation in RBFNN subspace is an approximation of motion representation in Isomap embedding subspace. For a new motion previously unseen, it can be simply mapped into the RBFNN subspace by the mapping function.

4 Separating Motion Style

4.1 Learning and Separating Style

Let the sets of input image sequences be $Y^k = \{y_i^k \in \mathbb{R}^o, i = 1, \dots, N_k\}$ and let their corresponding points on the unified embedding space be $\tilde{Y}^k = \{\tilde{y}_i^k \in \mathbb{R}^l, i = 1, \dots, N_k\}$ where l is the dimensionality of the embedding

space and $k=1\dots K$ is the style index. Let the set of N centers representing the mean manifold be $Av = \{Av_j \in R^l, j = 1, \dots, N\}$.

We can learn nonlinear mappings between the centers Av and each of the input sequence using generalized radial basis function interpolation GRBF [15], i.e., one mapping for each style class k .

Given learned nonlinear mapping coefficients C^1, C^2, \dots, C^K for each person, the style parameters can be decomposed by fitting an asymmetric bilinear model [19] to the coefficient tensor. Let the coefficients be arranged as a $o \times M \times K$ tensor C , where $M = (N + l + 1)$.

Therefore, we are looking for a decomposition in the form

$$C = F^c \times_3 B^s$$

where F^c is $d \times M \times J$ tensor containing content bases for the RBF coefficient space and $B^s = [b^1 \dots b^K]$ is a $J \times K$ style coefficients. This decomposition can be achieved by arranging the mapping coefficients as a $dM \times K$ matrix as

$$C = \begin{pmatrix} c_1^1 & \dots & c_1^K \\ \vdots & \ddots & \vdots \\ c_M^1 & \dots & c_M^K \end{pmatrix}$$

where $[c_1^k, \dots, c_M^k]$ are the columns for RBF coefficients C^k . Given the matrix C style vectors and contents bases can be obtained by singular value decomposition as $C = USV^T$ where the content bases are the columns of US and the style vectors are the rows of V .

4.2 Solving for Style and Content

Given a new input $y \in R^o$, it is required to find both the content, i.e., the corresponding embedding coordinates $\tilde{y} \in R^l$ on the manifold, and the person style parameters b^s . These parameters should minimize the reconstruction error defined as

$$E(\tilde{y}^c, b^s) = \|y - F \times b^s \times \varphi(\tilde{y}^c)\|^2$$

Solving for style: If the embedding coordinate (content) is known, we can solve for style vector b^s . Given style classes $b^k, k = 1, \dots, K$ learned from the training data and given the embedding coordinate, the observation can be considered as drawn from a Gaussian mixture model centered at $F \times b^k \times \varphi(\tilde{y})$ for each style class k . Therefore, observation probability $p(y | k, \tilde{y})$ can be computed as

$$p(y|k, \tilde{y}) \propto \exp\{-\|y - F \times b^k \times \varphi(\tilde{y})\|^2 / (2\sigma^2)\}$$

Style conditional class probabilities can be obtained as

$$p(k|\tilde{y}, y) = p(y|k, \tilde{y})p(k|\tilde{y}) / p(y|\tilde{y})$$

where $p(y|\tilde{y}) = \sum_k p(y|\tilde{y}, k)p(k)$. A new style vector can then be obtained as a linear combination of the k class style vectors as $b^s = \sum_k \omega_k b^k$ where the weights ω_k are set to be $p(k|\tilde{y}, y)$.

Given the two steps described above we can solve for both style b^s and content \tilde{y}^c in an EM-like iterative procedure where in the E-step we calculate the content \tilde{y}^c given the style parameters and in the M-step we calculate new style parameters b^s given the content. The initial content can be obtained using a mean style vector \tilde{b}^s .

5 Experimental Results

We implement our algorithm in matlab. It is more than 5,000 motion clips with 30 common types in the database for test. Most of the typical human motions are performed by actors, such as walking, running, kicking, punching, jumping, washing floor, etc.



Fig. 2. (Above) motion style generated in nonlinear subspace (below) in linear subspace

We compare the performance of our proposed nonlinear subspace motion synthesis algorithm with linear approach PCA subspace synthesis algorithm in Fig.2. As can be seen, Motion style generated in nonlinear subspace look more real than linear subspace and Isomap subspace outperforms PCA upon the motion data. Since the motion manifold is possibly highly nonlinear and PCA can only discover the linear structure. Table 1 shows that non-linear ISOMAP separating motion style has more higher success rate than linear PCA method.

Table 1. Comparison of success rate for separating motion style

	PCA	ISOMAP
walk	100%	100%
run	99.1%	99.3%
jump	78.2%	96.2%
boxing	50.1%	91.3%
dance	33.8%	90.1%

Fig.3 shows a series of key frames of three different motion type, one jump and two kinds of dancer. We can edit key frames to obtain a natural motion by adjust style parameters and content parameters in low-dimensional subspace, then a new motion include jump and dance is generated.

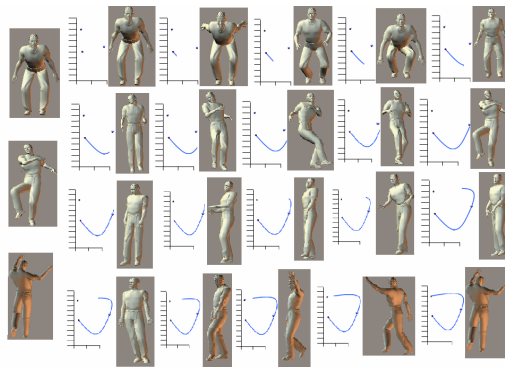


Fig. 3. Motion editing in nonlinear subspace

Then we adjust motion style parameters in nonlinear subspace to generate some exaggerated motion style for animation production. Some kinds of motion walk styles refer to human emotion are created, such as walk sadly, walk happily. These new motion styles cannot obtain by linear subspace motion synthesis and directly interpolation.

6 Conclusions

In this paper, an automatic motion synthesis and editing method is proposed. First, some motion features are extracted from motion data and generalization of Isomap with RBF neural network is used to reduce dimension of these features and embed original motion space and new motion data into low-dimensional non-linear subspace. we decompose the style and content parameters in the nonlinear subspace which map between a learned unified non-linear low-dimensionality subspace and the visual input space. According to adjust these parameters, new motion styles can be reconstructed.

References

1. Bruderlin, A., Williams, L.: Motion Signal Processing. In: ACM SIGGRAPH 1995, pp. 97–104 (1995)
2. Rose, C.F., Guenter, B., Bodenheimer, B., Cohen, M.F.: Efficient Generation of Motion Transitions Using Spacetime Constraints. In: ACM SIGGRAPH 1996, pp. 147–154 (August 1996)
3. Kovar, L., Gleicher, M.: Flexible Automatic Motion Blending with Registration Curves. In: ACM SIGGRAPH 2004, pp. 214–224 (2004)
4. Kovar, L., Gleicher, M., Pighin, F.: Motion Graphs. In: ACM SIGGRAPH 2002, pp. 473–482 (2002)
5. Rose, C., Bodenheimer, R., Cohen, M.F.: Verbs and Adverbs: Multidimensional Motion Interpolation Using Radial Basis Functions. *IEEE Computer Graphics and Applications* 18(5), 32–48 (1998)
6. Grochow, K., Martin, S.L., Hertzmann, A., Popović, Z.: Style-based Inverse Kinematics. In: ACM SIGGRAPH 2004, pp. 522–531 (2004)
7. Kovar, L., Gleicher, M.: Automated Extraction and Parameterization of Motions in Large Data Sets. In: ACM SIGGRAPH 2004, pp. 559–568 (2004)
8. Brand, M.E., Hertzmann, A.: Style Machines. In: ACM SIGGRAPH 2000, pp. 183–192 (2000)

Card Images Binarization Based on Dual-Thresholding Identification

Chunmei Liu¹, Duoqian Miao¹, and Chunheng Wang²

¹ Department of Computer Science and Technology, Tongji University, Shanghai, China

² The Key Laboratory of Complex Systems and Intelligence Science, Institute of Automation, Chinese Academy of Sciences, Beijing, China
chunmei.liu@mail.tongji.edu.cn

Abstract. In this paper, an algorithm is proposed for card images binarization. It is performed by three steps: coarse binarization, refined binarization, and postprocessing. Firstly, it uses the traditional global thresholding approach to separate a card image into several sub-images, which can be classified into two classes: text sub-images with clear background and text sub-images with complicated background. Secondly, the dual-thresholding is applied to regenerate or retouch the sub-images. According to the characteristics of text candidate sub-image, the thresholding method is selected and applied on it. Finally, the postprocessing is performed on the binary image. Experimental results demonstrate that this approach highly improved the performance of the card image binarization system.

1 Introduction

The automatic card recognition is an important area of document engineering, such as the recognition of passport, ID card, name card, mail image and so on. This application provides for document information portability. In automatic card recognition system, binarization is a crucial step to the success of subsequent recognition, which divides the image into two classes: the objects and the background. If poorly binarized images are used, card understanding would be difficult or even impossible. Compared with other image, in these card images, background patterns, noise characteristics, text quality, and histogram modalities are often unpredictable. This paper proposed a method to resolve the binarization of the card image with variable background patterns.

Thresholding is a simple and effective tool to separate objects from the background. Thresholding techniques can be categorized into two classes: one stage and multi-stage. One stage thresholding tries to find the good values at once. It can be further divided into two categories: global and local [11]. Global thresholding algorithms use a single threshold to separate objects from background. The most popular algorithms are Otsu's method [1], which chooses the threshold that minimizes within-group variance, Kittler and Illingworth's method [2], which chooses the threshold that minimizes Kullback information distance, and Huang and Wang's fuzzy binarization method [3]. The global methods are simple and runtime saving, and can

get the satisfying result when the document image is relative consistent. If the image is unevenly illuminated, the local thresholding might outperform the global method. The local thresholding algorithms compute a separate threshold for each pixel based on a neighborhood of the pixel, such as Niblack's method [5] which computed the local threshold from the mean and the standard deviation in each window, Bernsen's method [4], which computed the local threshold from the minimum and the maximum gray values, Sauvola method [6]. One of the limitations of local thresholding method is that it is often dependent on parameter.

The card image often has more complicated components, such as background patterns, noise, unpredictable text quality. One stage thresholding is not enough for it. In contrast, multi-stage thresholding methods are demonstrated that has good behavior in the binarization of images with complicated background.

The multi-stage thresholding techniques can utilize more information to extract the texts from the background [7-9]. Sue Wu [7] proposed a two stages thresholding method. The first stage is to use the global thresholding to get some children images. Then these children images undergo the selection whether the histogram follows a Gaussian normal distribution. If it follows the Gaussian distribution, the children image isn't selected to be performed the second stage thresholding. If it doesn't follow the Gaussian distribution, the second stage is performed on the children region by looking for the first minimum of histogram as thresholding value. This method made a success in the postal envelopes.

Solihin and Leedham [8] proposed a two stage thresholding approach for gray-scale handwriting images to separate the handwriting from the background. It classified the image into three classes: foreground, background, and a fuzzy area between them where it is hard to determine whether a pixel belongs to the foreground or the background. Native Integral Ratio (NIR) and Quadratic Integral Ratio (QIR) technique were presented based on this class. This method adapts to the handwriting image binarization.

In this paper, we propose an effective binarization approach for the card images with complicated background. Considering there are complex background pattern, lots of noise, and different text quality in the card images, we take the strategy of multi-stage thresholding. We can see from Fig.1 that the proposed approach is mainly performed by three steps: coarse binarization, refined binarization, postprocessing. In contrast to other approaches, in the first stage we use the traditional global thresholding approach to separate a card image into several sub-images. In the next stage we classify these sub-images into two classes: text sub-images with clear background and text sub-images with complicated background. We respectively analyze their different attributes and apply dual-thresholding to identify the classes which these text candidate sub-images belong to. According to the identification result, the local thresholding method can be appropriately selected for each text candidate sub-image. Finally, the postprocessing is performed to eliminate the non-text foreground by statistical analysis of text characteristic. Experimental results demonstrate that the proposed approach could efficiently be used as an automatic card image binarization system, which is robust for complicated background, texture background, noise, various text quality.

The paper is organized as follows. Section 2 presents the process of coarse binarization. Section 3 describes the individual step of refined binarization, and postprocessing is described in section 4. Section 5 demonstrates the experiment results and conclusions are provided in the final section.

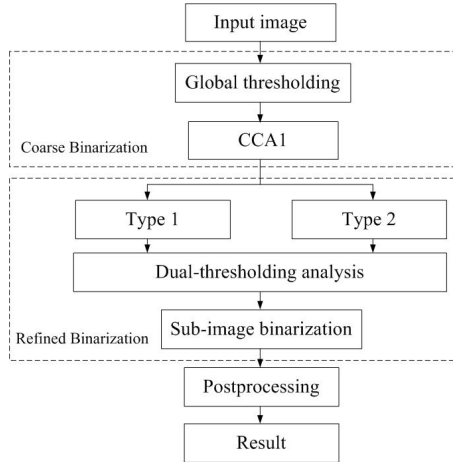


Fig. 1. Flow chart of the proposed approach

2 Coarse Binarization

This stage aims at segmenting the card image into regions for further processing. We begin with threshold method to get the preliminary binary image. Any thresholding technique may be used in this step since the primary purpose is to remove some simple background. Comparing with local adaptive threshold method, we focused on global thresholding method which is parameter independent and computationally inexpensive. Here we chose Otsu's method because it is simple and has been cited an effective scheme.

After coarse binarization on the entire image, the connected component analysis (CCA) is done on the resulting binary image to segment the image into n connected component regions. The position of each connected components is projected on original image. Then we get n sub-images: $FirR_1(1), FirR_1(2), \dots, FirR_1(n)$, which represent the text candidate regions, and the non-CC region is eliminated as the background region.

3 Refined Binarization

The objective in this stage is to find an optimal threshold for each sub-image that would eliminate the background noise, while preserving as much text strokes as possible.

3.1 Text Candidate Sub-images

After coarse stage, we acquired the sub-images that represent text candidates. Generally these sub-images can be classified into two classes: text images with clear background, text images with texture background.

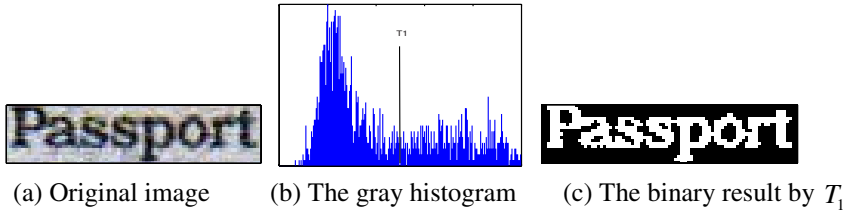


Fig. 2. Example of a text sub-image with clear background

The gray level histogram of card image usually has a valley and two peaks. Fig.2(a) is a text sub-image with clear background and Fig.2(b) is its gray level histogram. The histogram usually has a broad and low-flat valley and two clear peaks. The left peak of the gray level histogram is classified as background and the right one foreground. The histogram distribution is ideal, so it is easy to binarize it by a thresholding.

Fig.3(a) is a text sub-image with texture background and Fig.3(b) is its gray level histogram. It is difficult to partition the text foreground from the background because

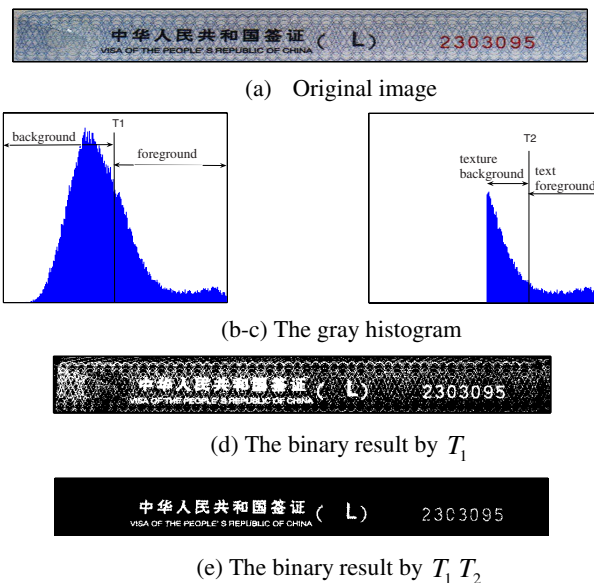


Fig. 3. Text sub-image with texture background

the texture background disturbs the histogram distribution. Here we analyze it by dual-thresholding. Firstly, we use the global thresholding to get the foreground, which not only includes the text parts but also the texture background. So in the next we project the position of the foreground on original image, seen from Fig.3(d), and compute the histogram of the projected parts, seen from Fig.3(c). Here we looked the text parts as foreground, and the texture parts as background. We use the global thresholding again on the projected parts. Then the final binarization result can be acquired, seen from Fig.3(e).

3.2 Dual-Thresholding

According to the characteristics of text candidate sub-images, we applied the dual-thresholding to regenerate the new sub-images or retouch the previous sub-images, and for the convenience called these regenerated and retouched sub-images as new sub-images. The dual-thresholding is performed by five steps.

Step1. Compute $T_1(i)$ by Otsu's thresholding method on sub-image $FirR(i)$, and get corresponding binary sub-image $FirB(i)$;

Step2. Project the position of $FirB(i)$ on the sub-image $FirR(i)$, and get the sub-image $SecR(i)$ for the second thresholding;

Step3. Compute $T_2(i)$ by Otsu's thresholding method again on the sub-image $SecR(i)$, and get corresponding binary sub-image $SecB(i)$;

Step4. Apply the CCA on the sub-image $SecB(i)$, and get new connected component region $SubR(1i)$, $SubR(2i)$, ..., $SubR(mi)$ of sub-image $SecR(i)$;

Step5. Decide the thresholding by the following two rules:

Rule1. when $m=1$, if width or height of $SubR(1i)$ is smaller than $FirR(i)$'s, $NewR(mi) = SubR(i)$, otherwise $NewR(mi) = FirR(i)$;

Rule2. when $m>1$, $NewR(i+1)$, $NewR(i+2)$, ..., $NewR(i+m) = SubR(1i)$, $SubR(2i)$, ..., $SubR(mi)$.

So the new sub-images are acquired: $NewR(1)$, $NewR(2)$, ..., $NewR(N)$, where $N = m1 + m2...mn$.

3.3 Text Sub-image Thresholds

Any thresholding technique which fits the sub-image can be used in this step. In this paper, for the comparison of experiment we apply Otsu's method on all the new sub-images. Namely if the text candidate sub-image has clear background, we select T_1 as thresholding of this subimage. If text candidate sub-image has texture background, we select T_2 as the thresholding of this sub-image.

3.4 Postprocessing

In this step firstly all the binary sub-images of the new sub-images are added to form the final binary image of the original card image.

After binarization there are still some non-text parts in the card image, such as portrait in passport and ID card, stamp on envelop. The size of non-text part often is larger than the character's. Here we compute the average size of character, and eliminate the non-text part if its size is much larger than the average size. At the same time the small part is removed as the noise. Thus the final binary image is acquired. Fig.5 shows the final result.

4 Experiment

In this section, in order to evaluate the effectiveness of the proposed method several experiments are carried out on 26 card images, which include passports, name cards, ID cards. These images contain variety of layouts, complicated background, different color texts, different text sizes.

4.1 Comparative Experiment

A comparative experiment is conducted between the proposed method and which are Otsu's method. Fig.4(a) shows an original passport image that has complicated background and text characters with different size and different grayscale. Fig.4(b) show the result of Otsu's method. The methods can not provide satisfactory recognition performance because of the disturbance of the complicated background. Fig.4(c) show the result by the proposed method. The result demonstrates the proposed algorithm can effectively remove the complicated background.

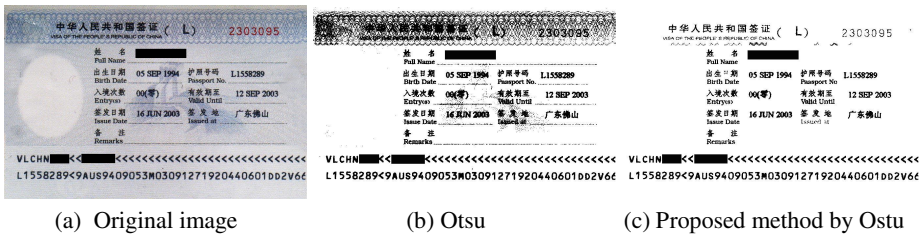


Fig. 4. Comparative experiment result between Otsu and Proposed method by Otsu

4.2 Comparative Experiment with Other Thresholding Methods

A comparative experiment is conducted between the proposed method and several other well-known methods, which are local entropy (LE), Niblack, Bernsen, and White method. In the proposed method, any thresholding technique can be used for the sub-image thresholding. Fig.5(a,c,e,g) respectively show the results of Local Entropy method (LE), Niblack, Bernsen, White method. Fig.5(b,d,f,h) show the results by the different sub-image thresholding methods which are Otsu's method, LE, Niblack, Bernsen, White method. The results demonstrate the proposed algorithm can effectively improve the performance of the binarization system.



(a) LE



(b) Proposed method by LE



(c) Niblack (150x150)



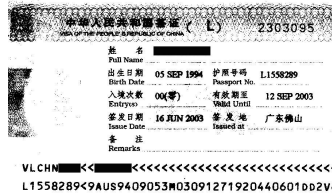
(d) Proposed method by Niblack



(e) Bensen (150x150)



(f) Proposed method by Bensen



(g) White (150x150)



(h) Proposed method by White

Fig. 5. Experiment result

5 Conclusion

In this paper, we have proposed an approach for the card image binarization, which uses the global information to perform coarse binarization and uses the local information to refine binarization. It uses the global information to acquire the text candidate sub-images. The dual-thresholding is applied to identify whether these text candidate sub-images have complicated background. According to the identification result, the local thresholding method can be appropriately selected for the text candidate sub-image, which makes the whole binarization have the perfect performance. The

experiment on the different card images shows that the proposed approach highly improved the performance of the card image binarization system.

In the future research, some problems need to be tackled. One is the thresholding method selection of the first stage, which directly influences the performance of the second stage. The other is the thresholding method selection of each sub-image, which can lead to the improved performance of the local binarization. Furthermore, when we apply this method on the card images, it is found that the proposed method has difficulties in binarizing the texts which have the approximate gray value with their surrounding background. In this case it is a good way to add color information. These problems need to be tackled in the future research.

Acknowledgments. This work was supported by Program for Young Excellent Talents in Tongji University.

References

1. Otsu, N.: A Threshold Selection Method from Grey Level Histogram. *IEEE Transactions on Systems, Man and Cybernetics* 9(1), 62–66 (1979)
2. Kittler, J., Illingworth, J.: Minimum Error Thresholding. *Pattern Recognition* 19, 41–47 (1986)
3. Huang, L.K., Wang, M.J.: Image Thresholding by Minimizing the Measure of Fuzziness. *Pattern Recognition* 28, 41–51 (1995)
4. Bernsen, J.: Dynamic Thresholding of Grey-level Images. In: 8th International Conference on Pattern Recognition, France-Paris, pp. 251–255. IEEE Computer Society, Los Alamitos (1986)
5. Niblack, W.: An Introduction to Image Processing, pp. 115–116. Prentice-Hall, Englewood Cliffs (1986)
6. Sauvola, J., Pietikainen, M.: Adaptive Document Image Binarization. *Pattern Recognition* 33, 225–236 (2000)
7. Wu, S., Amin, A.: Automatic Thresholding of Gray-level Using Multi-stage Approach. In: 7th International Conference on Document Analysis and Recognition, vol. 1, pp. 493–497. IEEE Computer Society, Washington (2003)
8. Solihin, Y., Leedham, C.: Integral Ratio: A New Class of Thresholding Techniques for Handwriting Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 21, 761–768 (1999)
9. Pal, N.R., Pal, S.K.: Entropic Thresholding. *Signal Process* 16, 97–108 (1989)
10. Du, Y., Chang, C., Thouin, P.D.: An Unsupervised Approach to Color Video Thresholding. *Optical Engineering* 43(2), 282–289 (2004)
11. Sezgin, M., Sankur, B.: Survey over Image Thresholding Techniques and Quantitative Performance Evaluation. *Journal of Electronic Imaging* 13(1), 146–165 (2004)
12. Wang, S.Z., Lee, H.J.: Dual-Binarization and Anisotropic Diffusion of Chinese Characters in Calligraphy Documents. In: 6th International Conference on Document Analysis and Recognition, pp. 271–275. IEEE Computer Society, Washington (2001)

Defects Identification in Textile by Means of Artificial Neural Networks

Vitoantonio Bevilacqua^{1,2}, Lucia Cariello^{1,2}, Giuseppe Mastronardi^{1,2},
Vito Palmieri¹, and Marco Giannini³

¹ Department of Electrical and Electronics, Polytechnic of Bari,
Via Orabona, 4 – 70125 Bari, Italy
{Bevilacqua, Mastronardi}@poliba.it, Cariello@deemail.poliba.it,
nicpalm@libero.it

² e.B.I.S. s.r.l. (electronic Business in Security), Spin-Off of Polytechnic of Bari,
Str. Prov. per Casamassima Km. 3 – 70010 Valenzano (BA), Italy

³ Centro Laser s.c.a.r.l.,
Str. Prov. per Casamassima Km. 3 – 70010 Valenzano (BA), Italy
marco.giannini@centrolaser.it

Abstract. In this paper we use a neural network approach for defects identification in textile. The images analyzed came from an artificial vision system that we used to acquire and memorize them in bitmap file format. The vision system is made of two grey scale line scan camera arrays and each array is composed of four CCD cameras with a sensor of 2048 pixels. Every single camera has a field of view of 600mm. The big amount of pixels to be studied to determine whether the texture is defective or not, requires the implementation of some encoding technique to reduce the number of the significant elements. The artificial neural networks (ANN) are manipulated to compress a bitmap that may contain several defects in order to represent it with a number of coefficients that is smaller than the total number of pixel but still enough to identify all kinds of defects classified. An error back propagation algorithm is also used to train the neural network. The proposed technique includes, also, steps to break down large images into smaller windows or array and eliminate redundant information.

Keywords: texture defects identification; vision system; artificial neural networks; error back propagation algorithm

1 Introduction

The analysis of images in an artificial vision system is an expensive process. Image compression provides an option for reducing the complexity and the processing time for defects identification. Artificial Neural Networks (ANNs) have been applied to many problems, and have demonstrated their excellence over traditional methods. One example of such application is the compression of images due to their superiority when dealing with noisy or incomplete data. Artificial Neural networks, in fact, seem to be well suited to image compression, as they have the ability to preprocess input

patterns to produce simpler patterns with fewer components. This compressed information preserves the full information obtained from the external environment. These networks contain at least one hidden layer, with fewer units than the input and output layers. The Neural Network is trained to recreate the input data. Its bottleneck architecture forces the network to project the original data onto a lower dimensional manifold.

There have already been an exhaustive number of published papers showing the application of ANNs to image compression [1-6]. Many different training algorithms and architectures have been used. Some of the more notable in the literature are: nested training algorithms used with symmetrical multilayer neural networks [6], self organizing maps for codebook generation [2], principal component analysis networks [1], back propagation networks [5], and the adaptive principal component extraction algorithm [4]. In particular, the Back Propagation Neural Network Algorithm performs a gradient-descent in the parameters space, minimizing an appropriate error function. The weights update equations minimize this error.

The purpose of this paper is to present a method to identify defects in textile images using the artificial neural network compression approach based on error back propagation algorithm. Starting from the images of textile, acquired by the artificial vision system, we made a classification of defects based on common characteristics and properties, in particular:

- Defects along columns such as contaminations with other materials or wireless
- Defects along raw such as problems in texture in which there are two wire in the same location
- Circular defects in an area of textile with lacerations or hole, such as knots and tears

2 The Vision System

The Vision System implemented for the automatic inspection of textile is constituted of two different arrays of greyscale line scan camera, as described in the introduction.

The first array analyzes structural defects of textile. To emphasize them a backlight source is used. The light source is made by two fluorescent high brightness tubes, while the alimentation is provided by an high frequency generator. The reason behind the choice of the high frequency generator resides in the necessity of producing an intensity of illumination that should be uniform during the acquisition of images.

Thanks to the backlight illumination the detection of holes, lacking wires or two - tired wires is made easier. The system locates those kind of defects because the image will present regions with an high contrast when compared to the background (i.e. darker regions for two-tired wires, lighter for the other defects).

The light system that conditions the image acquisition of the second array of cameras is constituted of direct halogen lamps.

The incident illumination is helpful to analyze dark or thick tissue. This kind of illumination highlights leaning defects like nodes which result emphasized by the edge top light. A panel is interposed between the two benches to prevent light interferences.

For each camera an automatic system that controls and rectifies the Shutter (exposition time length of the camera) is implemented. This control ensures the acquisition of images with a default grey level. The grey level target is 127 on a scale from 0 to 255, in order to guarantee the maximum oscillation range for both white and black spots.

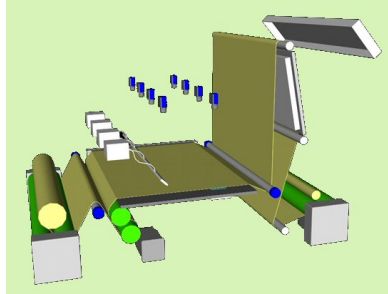


Fig. 1. Representation of the acquisition stage of the prototype

The analysis software is implemented on four Pc platforms equipped with frame-grabber boards. Each Pc drives a couple of cameras and executes the software that performs the defect detection. The results are collected and sent to a supervising platform through an Ethernet network. The supervisor hosts a database where the results of the inspection are stored and controls the motor's speed.

The Vision System receives the acquisition trigger from an encoder that is connected to a motorized roll.

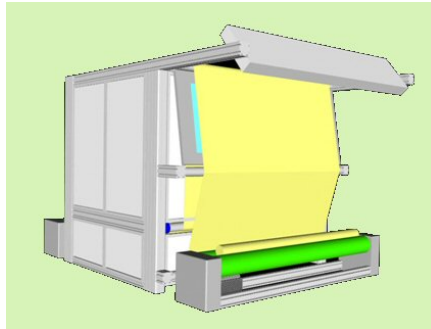


Fig. 2. Schematic representation of the prototype

The mechanic structure of the machine is constituted of a box containing cameras and illumination system, an automatic position system that provides the alignment of the tissue during the wrapping phase and a system of motorized cylinders that are in charge for the unwrap and the subsequent wrap of the tissue.

The movement system is realized using three brushless motors that lead three rolls: one that wraps the tissue, another one that unwraps it and the last one that stretches

the tissue to guarantee the cameras the best acquisitions conditions. For this purpose four load cells are installed to verify the tension of tissues and to avoid folds.

If the image analysis requires long time, the supervisor slows down the progression speed of the tissue.

Before the final wrapping of the tissue, there is a panel where the manual inspection of textile can be performed. This machine's side has been useful during the training time for a real time verification of the results given by the Vision System.

3 Artificial Neural Network Based on Error Back Propagation

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way of the biological nervous systems process information. The key element of this paradigm is the novel structure of the information processing system.

It is composed of a large number of highly interconnected processing elements (neurons) cooperating to solve specific problems. All connections among neurons are characterized by numeric values (weights) that are updated during the training. If we consider n the number of neurons in the input layer, m the number of neurons in the output layer, N_l the number of neurons belonging to the l^{th} layer and $o_k^{(l)}$ be the output of the k^{th} neuron of the l^{th} layer, then the computation performed by each neuron can be expressed as:

$$net_k^{(l)} = \sum_{j=1}^{N_{(l-1)}} w_{kj}^{(l)} o_j^{(l-1)} \quad (1)$$

$$o_k^l = f(net_k^l) \quad (2)$$

where $net_k^{(l)}$ is the weighted sum of the k neurons of the l^{th} layer, $w_{kj}^{(l)}$ is the weight by which the same neuron multiplies the output $o_j^{(l-1)}$ of the j^{th} neuron of the previous layer and $f(\cdot)$ is a nonlinear bounded function, often the sigmoid function. The ANN is trained by a supervised learning process: in the training phase the network processes all the pairs of input-output presented by the user, learning how to associate a particular input to a specific output trying to extend the information acquired also for cases that don't belong to the *training set spectrum*. Any pair of data in the training set is presented to the system a quantity of time determined by the user *a priori*.

For this work an ANN is used to compress a bitmap image in order to represent it with a number of coefficients that is smaller than the total number of pixels to defects identification in textile. The learning step is based on the Error Back Propagation (EBP) algorithm [7].

This algorithm is based on the error correction learning rule. Error propagation consists of two passes through the different layers of the network, a forward pass and a backward pass. In the forward pass the input vector is applied to the network sensory nodes and its effect propagates through the network layer by layer. Finally a set of outputs is produced as the actual response of the network. During the forward pass the synaptic weights of the network are all fixed. In the back pass the synaptic weights are updated sequentially, from the output layer back to the input layer, by propagating an error signal backward along the neural connections (hence the name "back-propagation") according to the gradient-descent learning rule:

$$\Delta w_{ij} = -\gamma \frac{\partial E}{\partial w_{ij}} \quad 0 < \gamma < 1 \quad (3)$$

While using the coefficients obtained from ANN instead of the pixels, we get a very low system computational complexity that allows, in real-time, to identify and to detect the exactly position of a textile problems.

4 Experimental Results

The ANN, using EBP algorithm, extracts, in our work, the main features from the 256x256 pixel images, to store them in a sequence of:

- 256 bits for defects along row or column;
- 127x127 dimension window for circular defects;

reducing in this way the computational complexity of the problem.

The used neural network, according to the type of defect to analyze and identify, has been designed with different characteristics (in particular as regards the number of input layer - hidden layer - output layer neurons) but the same basic structure, so that it can adapt itself to the nature of the defect.

4.1 Row/Column Defects Detection

The image that represent the portion of textile under test is divided in row or column representing the input for the network that search for defects along row/columns.

The first ANN layer is formed by 256 neurons which are the rows or columns values (one neuron per pixel); the hidden layer is composed by only 1 neuron (that returns an output value for each row/column with values ranging from zero to one, as depicted in Fig. 3) and the last layer is formed by the same number of neurons of the input layer. The procedure depicted in Fig. 3 has been repeated for all image rows/columns in order to obtain 256 values. In this way it is possible to gain a “strips” image, rebuilt by image coefficient vectors.

The ANN has been trained by using an equal input and output training set and equal only to rows/columns white or black, in order to train the net on the two possible maximum contrasts present in an input image. At the start of training, the error is high, but then it decrease, reaching, at the end of the process, an asintotic value (see the learning curve in Fig. 4).

In the learning phase, the network began to specialize too much on training set but lose generalization.

To test the capacity of generalization of ANN during the training is carried out periodically the classification of a set of samples not belonging to training set (validation set). The trend of learning is shown by two curves, that of the training set and that of validation set (Fig. 5). The stop of the training phase is corresponding to the minimum on the validation set curve.

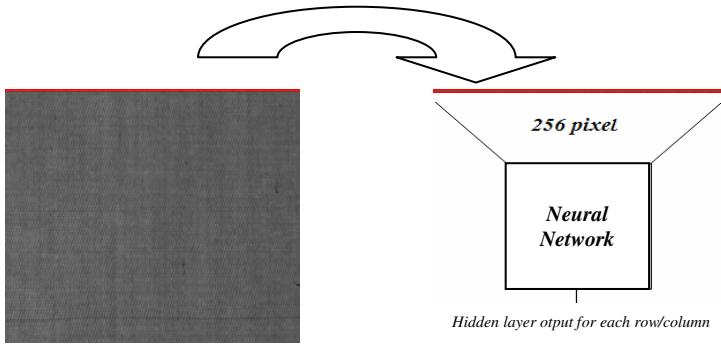


Fig. 3. Schematic representation of the hidden layer output in row/column defects detection

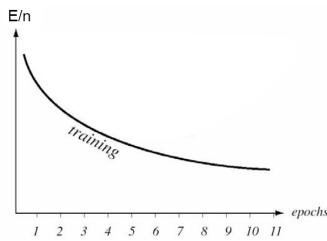


Fig. 4. Learning curve

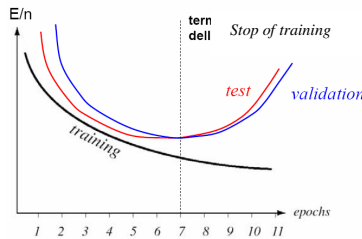


Fig. 5. Learning curve and validation curve

After the training phase, the network is able to work as a pure linear function, where the input of the first layer must be the same of the output of the last layer.

So, for every image we have an array of 256 values that characterize each row/column. The matrix of weights referred to the connections between the inputs and the hidden layer codifies the image bitmap, while the matrix of weights associated to the connections between the hidden layer and the outputs decodes the sequence of bits. At this point, the matrix of weights is stored and finally the ANN is tested with images containing defects that are different from the training set features.

Finally, if we have the image in Fig. 6 (a), the corresponding “strips” image is depicted in Fig. 6 (b). Moreover the results can be graphically valued as in Fig 7, where in 231 and 232 rows there are two crests that identify the most pronounced rows in the original image (see red arrow in Fig. 6 (b)).

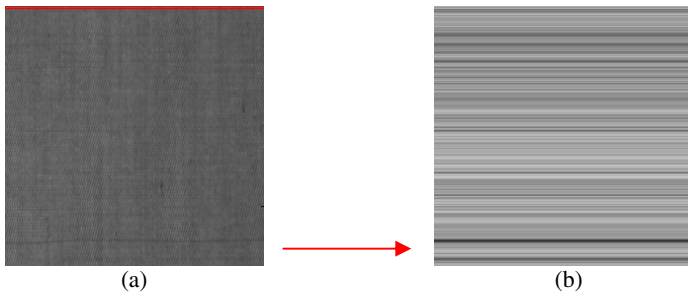


Fig. 6. Original image (a). The obtained “strips” image (b).

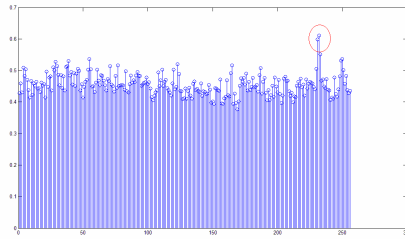


Fig. 7. Chart of the outputs for each row/column

4.2 Circular Defects Detection

In the detection of circular defects, instead, the input of the network is an image sub matrix of 4x4 elements (16 pixel) that scans completely the image with a superimposition of 2 pixel moving in the raw or in the column.

The first ANN layer, in this case, is formed by 16 neurons, which are the values of 4x4 windows (one neuron per pixel); the hidden layer is composed by only 1 neuron (that returns an output value for each sub matrix with values ranging from zero to one as depicted in Fig. 8) and the last layer, even, by 16 neurons.

The steps described in Fig. 8 have been repeated until the entire image has been scanned by windows in order to obtain an 128x128 pixel image of windows, where

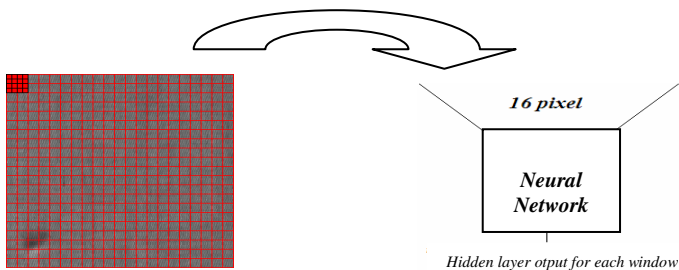


Fig. 8. Schematic representation of the hidden layer output in circular defects detection

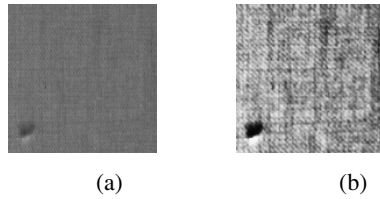


Fig. 9. Original image (a). The obtained “windows” image (b).

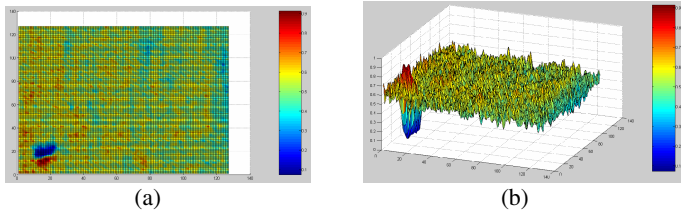


Fig. 10. 2D chart of the outputs (a). 3D chart of the outputs (b).

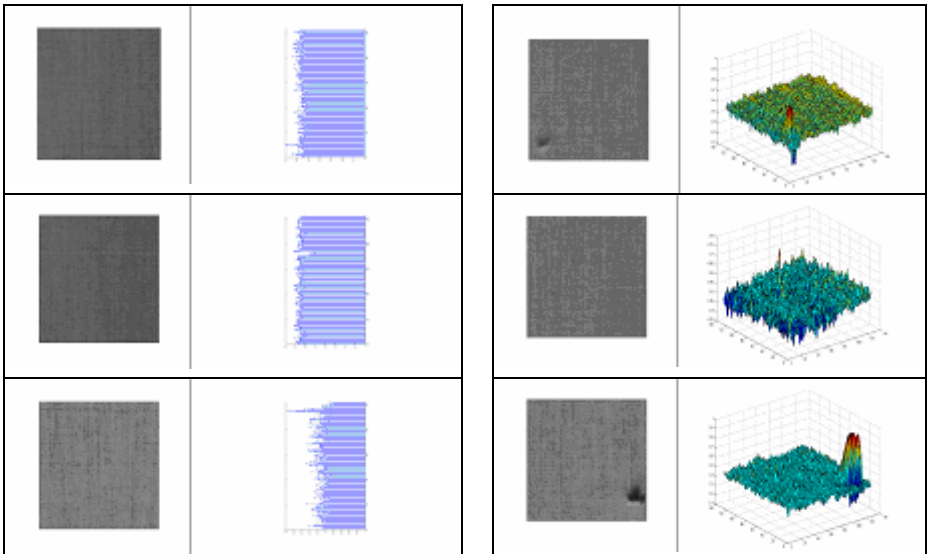


Fig. 11. Examples of row/column defects **Fig. 12.** Examples of circular defects detection

each of them represents the hidden neuron output brightness value of the corresponding window, multiplied by 255. In this way it is possible to gain a “windows” image, rebuilt by image coefficient vectors.

The ANN has been trained using white or black windows in order to train the net with the maximum possible contrast present in an input image. The training phase and the stop of learning process is the same described in the previous subsection. After

the training phase, the compressed image is described by 128x128 bits, that are the outputs of a hidden layer consisting of a sigmoid function processing elements.

Finally, if we have the image in Fig. 9 (a), the corresponding “windows” image is depicted in Fig. 9 (b). Moreover the results can be valued by the two charts in Fig 10 (a) and (b): the points nearer to value one (red points) represent the not homogeneous clearest image parts, while the points nearer to value zero (blue points) represent the not homogeneous darkest image parts (see also Fig. 9 (b)). Others results are depicted in Fig 11. e Fig. 12.

5 Conclusions

In this paper a neural network approach for Row/Column and Circular defects identification in textile has been proposed. An ANN, trained by an error back propagation algorithm, is used to represent the images with a number of coefficients that is smaller than the total number of pixel but enough to identify the defects.

References

1. Oja, E.: Data Compression, Feature Extraction, and Autoassociation in Feedforward Neural Networks. In: Kohonen, et al. (eds.) *Artificial Neural Networks*, pp. 737–745. Elsevier Science Publishers, Amsterdam (1991)
2. Dony, R.D., Haykin, S.: Neural Network Approaches to Image Compression. *Proceedings of the IEEE* 23(2), 289–303 (1995)
3. Carato, S.: Neural Networks for Image Compression. In: Gelenbe, E. (ed.) *Neural Networks Advances and Applications*, vol. 2, pp. 177–198. Elsevier Science Publishers, Amsterdam (1992)
4. Kung, S.Y., et al.: Adaptive Principal Component Extraction (APEX) and Applications. *IEEE Transactions on Signal Processing* 42(5), 1202–1217 (1994)
5. Mougeot, M., et al.: Image Compression with Backpropagation: Improvement of the Visual Restoration Using Different Cost Functions. *Neural Networks* 4, 467–476 (1991)
6. Namphol, A., et al.: Image Compression with a Hierarchical Neural Network. *IEEE Transactions on Aerospace and Electronic Systems* 32(1), 327–337 (1996)
7. Haykin, S.: *Neural Networks-A Comprehensive foundation*. Pearson Education edn. (2004)

Diabetic Damage Detection in Retinal Images Via a Sparsely-Connected Neurofuzzy Network

Leonarda Carnimeo

Dipartimento di Elettrotecnica ed Elettronica, Politecnico di Bari, via E.Orabona,
4-70125 Bari, Italy
carnimeo@deemail.poliba.it

Abstract. In this paper a contribution towards diabetic damage detection in retinal images is proposed by synthesizing a Sparsely Connected Neurofuzzy Network for fundus image processing in the presence of retinopathies. A Hopfield-like neurofuzzy subnetwork is firstly synthesized to obtain contrast-enhanced images. After an optimal thresholding performed by an MLP-based neural subsystem, contrast-enhanced images are then globally segmented by a further sparsely-connected neural subnet to highlight vague pale regions. In this way diabetic damaged areas reveal isolated in bipolar output images. Experimental cases are reported and discussed.

1 Introduction

In ophthalmic field specific retinal damages, called *exudates*, which appear as vague pale areas in fundus images, can reveal diabetic retinopathies [1], [2]. In the last years contributions aiming at the development of diagnostic tools for an easier identification of diabetic symptoms in fundus images were proposed [3-5], as well as approaches to the detection of such symptoms using neural networks were suggested in [6-8]. In [6] MLP neural networks were trained for classifying diabetic symptoms and, due to the vagueness of information in medical images, in [7] fuzzy processing systems began to be applied in ophthalmic studies. Furthermore, in [8] a network of spiking neurons was considered to detect symptoms in segmented retinal images.

In this paper, a contribution for improving an intelligent detection of diabetic exudates is presented by synthesizing a Sparsely-Connected Neurofuzzy Network (SCNN) for retinal image processing. The sparsely-connected neural structure with fixed or adjustable weights makes this kind of networks very suitable for a VLSI implementation. After introducing an adequate image processing procedure of the whole neural system and taking into account results in [9], a Hopfield-like neurofuzzy subnetwork is herein designed to highlight vague pale regions in diabetic fundus images by an effective neurofuzzy synthesis technique. A subnetwork of Multi-Layer Perceptron neural networks is successively trained for an optimal thresholding, with the aim of minimizing pixel classification errors. Contrast-enhanced images are then segmented via a further sparsely-connected neural subnet, providing bipolar output images, in which suspect diabetic damaged areas are easily isolated. Performances of the whole Neurofuzzy Network are evaluated. The capabilities of the proposed system are illustrated and discussed by means of experimental cases.

2 Diabetic Damage Detection Via a SCNN

An *ad hoc* procedure for retinal image analysis is herein introduced, mainly based on three processing tasks: a) Neurofuzzy Contrast Enhancement; b) NN-based Optimal Thresholding; c) Global Segmentation. These steps are performed by the proposed neurofuzzy system as reported in Fig.1.

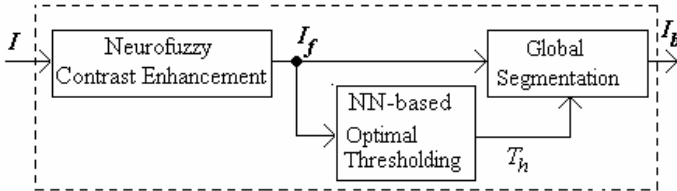


Fig. 1. Processing tasks of the diabetic damage detection procedure

The input image I the green layer of an RGB retinal image. Vague pale areas, suspected to be diabetic symptoms, have to be detected in image I . Thus, an adequate image segmentation has to be carried out in order to segment each *fundus* image in two *Suspect/Not-Suspect* sets, each one supposed as distinguishing a clinically significant area. Due to vagueness of pale regions and to the strong nonlinearity of histograms in fundus images I , an optimal thresholding can be computationally effective if histograms of analyzed images reveal bimodal, that is, the two sets to be identified have the maximum distance in brightness space. This can be done via a neurofuzzy image contrast enhancement network. A sparsely-connected neural subnet NN_F , behaving as a fuzzy system, is herein implemented to process each image I and obtain a contrast-enhanced image I_f with a bimodal histogram (Fig.2).

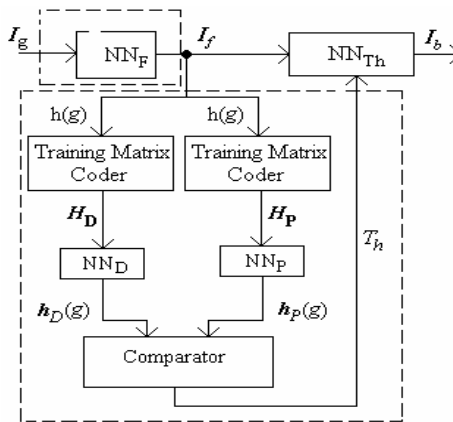


Fig. 2. Architecture of the Sparsely-Connected Neurofuzzy Network

Then, an optimal thresholding is developed by means of two MLP neural networks behaving as interpolators of histogram values. Finally, a global bipolar segmentation is performed on images I_f via a time-invariant sparsely-connected neural subnet NN_{Th} , providing binary output images in which diabetic damages are highlighted..

2.1 Neurofuzzy Contrast Enhancement

As is well known, contrast enhancement of digital retinal images should be achieved preserving anatomic details. For this purpose, a Hopfield-like Neurofuzzy Network NN_F is developed, by defining two fuzzy antecedent sets and two fuzzy consequent ones adequate to describe the semantic content of retinal images [9]. In detail, each fundus image I is considered as composed by two partially overlapped input fuzzy subsets, called *Deep D* and *Pale P*, respectively, and defined as:

$$\begin{aligned}
 \mathbf{D} &= \{D(g) = m_D(g) \mid 0 \leq g \leq b\} \\
 \mathbf{P} &= \{P(g) = m_P(g) \mid a \leq g \leq 255\} \quad 0 \leq a < b \leq 255
 \end{aligned}$$

being $g = 0, \dots, 255$ the generic grey value of each pixel. Right-angled triangular membership functions are adopted for $m_D(g)$, $m_P(g)$ with values in $[0; 1]$. In an analogous way, the domain of output values in $[-1; 1]$ is quantized into two output *Not-Suspect/Suspect NS/S* fuzzy subsets, respectively. Basing on the assumption that pale areas represent suspect retinal damages, the fuzzy rules which provide a proper mapping from input images into output contrasted ones can be expressed as:

$$\begin{aligned}
 \text{IF } p_{ij} \in \mathbf{D} \text{ THEN } f_{ij} \in \mathbf{NS} \\
 \text{IF } p_{ij} \in \mathbf{P} \text{ THEN } f_{ij} \in \mathbf{S}
 \end{aligned}$$

where p_{ij} and f_{ij} denote grey level values of each pixel in input images I and in contrast-enhanced ones I_f , respectively. All possible values that a generic pixel p_{ij} can assume in $[0, 255]$ are codified by considering

$$\mathbf{x}_g = [m_D(g) \ m_D(g) \ m_P(g) \ m_P(g)] = [x_{g1} \ x_{g2} \ x_{g3} \ x_{g4}] \in \mathbb{R}^{1 \times 4}$$

The reported fuzzy rules can be encoded by a (4x2)-matrix \mathbf{F} as in [9]. An output vector $\mathbf{y}_g = [y_{g1} \ y_{g2}] \in \mathbb{R}^{1 \times 2}$ can be associated to each input \mathbf{x}_g by $\mathbf{y}_g = \mathbf{x}_g \otimes \mathbf{F}$, where each input vector \mathbf{x}_g contains the values of membership functions for $g = 0, \dots, 255$. The generic component of vector \mathbf{x}_g can assume one among 256 fuzzy values in $[0, 1]$ for each grey level of image I . Moreover, the components of the output vector \mathbf{y}_g can assume only two fuzzy values corresponding to the degree of membership to the subsets $\mathbf{NS/S}$, respectively. The elements y_{g1} and y_{g2} are inferred with the well-known method of the centre of gravity to obtain a fuzzy value f_{ij} for each value of p_{ij} . For example, if $p_{ij} = \hat{g}$, then the grey level value of f_{ij} in image I_f is given by:

$$f_{ij} = \frac{255}{2} \left(\frac{-y_{\hat{g}1} + y_{\hat{g}2}}{y_{\hat{g}1} + y_{\hat{g}2}} + 1 \right)$$

where $\mathbf{y}_{\hat{g}} = [y_{\hat{g}1} \ y_{\hat{g}2}]$. In this way, the histogram of I_f is emphasized toward the extreme values of gray levels with respect to image I and a contrast enhancement can be achieved. A (4x4)-neuron neurofuzzy subnetwork with neighborhood of unit order, can now be synthesized. The training data set has to be formed by input/output vector pairs $(\mathbf{x}_g; \mathbf{y}_g)$ which codify the previously defined fuzzy rules. Each neuron of this sparsely-connected subnet is characterized by a piecewise linear output function. During the training phase, input/output vector pairs have to be submitted to this Hopfield-like neural subnet to be synthesized with the following feedback/feedforward templates

$$\mathbf{B} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \qquad \mathbf{A} = \begin{pmatrix} 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 0 \end{pmatrix}$$

and null bias vector. The interconnection weight matrix \mathbf{W} can now be determined following the synthesis procedure suggested in [9]. The synthesized sparsely-connected neural subnet behaves as a fuzzy system, able to enhance image contrast. Each output image I_f presents a bimodal histogram, given by

$$h(g): g \rightarrow h_g \quad g = 0, \dots, 255 \quad \text{being} \quad h_g = \text{cardinality}\{ (i, j) \mid I_f(i, j) = g \}$$

2.2 NN-Based Optimal Thresholding

After a neurofuzzy contrast-enhancement, processed images I_f present two-peak histograms. The first peak concerns with information about deep areas in images I , the other peak concerns with pale regions. Retinal damaged areas can be highlighted in each contrast-enhanced image I_f by an adequate global segmentation. For this purpose, an optimal thresholding is developed, by requiring that errors in classifying suspect regions be minimized. A neural subnetwork formed by two Multi-Layer Perceptron networks NN_D and NN_P is designed for an optimal thresholding as shown in Fig.2. For this purpose, let m denote the maximum gray level in $[1; 254]$ such that $h(m)$ is a relative minimum of histogram $h(g)$. By defining the vectors

$$\begin{aligned} \mathbf{g} &= [1, 2, \dots, 255]^T \in \mathbb{N}^{255 \times 1} \\ \mathbf{h}_D &= [h_1, \dots, h_k, \dots, h_m, 0 \dots, 0]^T \in \mathbb{N}^{255 \times 1} \\ \mathbf{h}_P &= [0, \dots, 0, h_{m+1}, h_{m+2}, \dots, \dots, h_{255}]^T \in \mathbb{N}^{255 \times 1} \end{aligned}$$

containing occurrences of deep/pale gray level values only, respectively, the following matrices

$$\mathbf{H}_D = [\mathbf{g}, \mathbf{h}_D] \in \mathbb{N}^{255 \times 2} \qquad \mathbf{H}_P = [\mathbf{g}, \mathbf{h}_P] \in \mathbb{N}^{255 \times 2}$$

containing informations about deep areas and pale ones of contrast-enhanced images I_f can be obtained. These matrices provide proper sets for training each neural network to recognize one mode of the bimodal histogram $h_j(g)$. The optimal threshold, is given by the value T_h such that [10]

$$h_D(T_h) = h_P(T_h)$$

Input layer and output layer of neural networks NN_D and NN_P each consists in one neuron with a logarithmic sigmoid transfer function. Such MLP networks have one hidden layer, whose optimal number of neurons can be computed by evaluating the minimum values of the

$$\text{Mean Maximum Error Index} \quad \text{MME} = \frac{1}{r} \text{Max} \{ |h_k(g) - h(g)| \}$$

and of the

$$\text{Mean Percentage Error Index} \quad \text{MPE} = \frac{100}{r} \sum_{l=1}^r \sum_{g=1}^{255} \frac{|h_k(g) - h(g)|}{h(g)}$$

for $k=P, D$ and $g = 1, \dots, 255$, being r the number of training phases carried out with different initial weights. Neural networks NN_D and NN_P behave as interpolators of histograms $h_D(g)$ and $h_P(g)$ by fitting discrete values in both modes which form histogram $h_f(g)$. Neural networks are then trained using the Levenberg-Marquardt back-propagation (LMBP) training algorithm [10]:

2.3 Global Segmentation

Global segmentation is the final step to provide output binary images containing only significant information about suspect damaged areas. Errors in suspect area detection can be minimized if an optimal threshold T_h , computed by the Optimal Thresholding Neural Subnetwork, is determined.

Thus, I_f can be segmented as follows

$$I_b(i, j) = \begin{cases} 255 & \text{if } I_f(i, j) < T_h \\ 0 & \text{if } I_f(i, j) \geq T_h \end{cases} \quad i = 0, \dots, M, \quad j = 0, \dots, N$$

where in the bipolar mask I_b black pixels identify diabetic damaged areas and optic disk of the original fundus image I . This template A is adopted for weights in NN_{Th}

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad \text{and a bias value given by } c = 1 - 2T_h/255$$

3 Results and Discussion

An interesting soft computing approach to the detection of diabetic symptoms was suggested in [6], where MLP neural networks were trained for classifying diabetic symptoms, but synthesized networks, besides effective, require heavy computational efforts. Furthermore, due to the vagueness of information in medical images, in recent years fuzzy techniques have also been involved for color fundus image segmentations [7], but proposed algorithms often reveal quite sensitive both to selective features and to color space representation. Moreover, in [8] a network of spiking neurons was considered to detect symptoms in segmented retinal images, but drawbacks arise for

high memory requirements and long computation times. In this work, the capabilities of the proposed Sparsely-Connected Neurofuzzy Network have been investigated on several (450x530) retinal images. In Fig.3 the green layer of a selected *fundus* image I and its histogram $h(g)$ are reported, where diabetic symptoms given by vague pale regions can be noted.

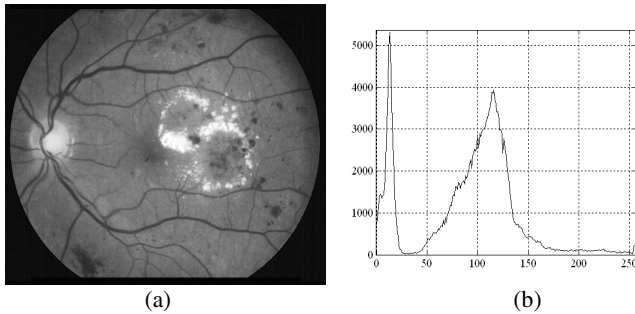


Fig. 3. (a) Input *fundus* image I ; (b) histogram $h(g)$

Following the suggested procedure, a (4x4)-cell sparsely-connected neural network has been implemented for the fuzzy values $a=25$ and $b=200$ which minimize MME and MPE. Fig.4 shows contrast-enhanced image I_f and its bimodal histogram obtained by processing the selected image I with the designed neurofuzzy subnet. By considering that an amount of close pale pixels indicates a suspect damaged area, the peak value in the histogram $h_f(g)$ corresponds to a bright suspect area.

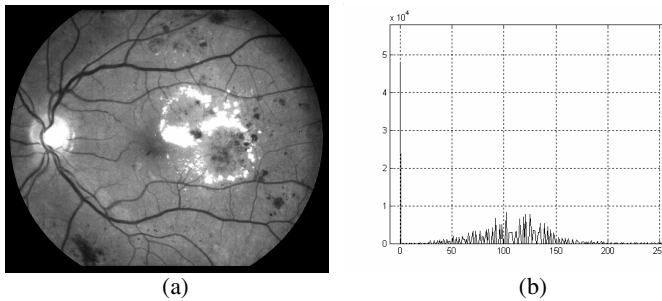


Fig. 4. (a) Contrast-enhanced image I_f ; (b) histogram $h_f(g)$

Both neural networks NN_D and NN_P with one hidden layer have been then trained by considering the LMBP learning algorithm. The optimal number of 9 hidden neurons has been determined for $r=20$ training phases by evaluating MME and MPE [10]. Moreover, due to the bimodal behaviour of the histogram $h_f(g)$ [4], an optimal thresholding can provide a value $T_h = 207$. In Fig.5 results are shown, where black pixels belong to regions that can be classified as suspect damaged areas.

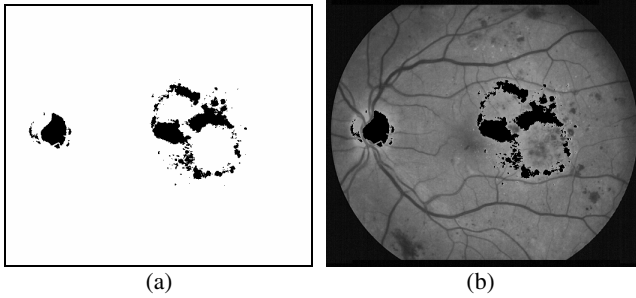


Fig. 5. (a) Output image I_b of the presented neural system; (b) Superimposition to image I

Performances of the proposed NN-based system are evaluated by comparing output images to *gold standard ones*, provided by clinicians. Results are discussed by determining the amount of True Positives TP, False Negatives FN and False Positive FP, as defined in [2]. A percentage measure of exactness is given by the *Correct Recognition Rate (CCR)* [4], being $(FN + FP)$ the number of misclassified pixels

$$CCR = 100 \left(1 - \frac{FN+FP}{\text{Total Number of pixels}} \right)$$

This index gives a percentage measure of correctly classified regions in comparison with known results, given by expert clinicians in *gold standard* images. The values of FP and FN have been computed by considering a (200×200) -window centered on the macula in image I_b , being macula the main ophthalmic region. A comparison of CCR% values obtained as in [2] and in [5] for image I_b is shown in Table 1.

Table 1. Comparison of CCR% values

	Method in [2]	Proposed Network	Method in [5]
CCR%	99.92	98.86	98.55

The synthesized Sparsely-Connected Neurofuzzy Network presents a value of $CCR\% = 98.86$. It can be noted that this value reveals better than the value of CCR% computed as in [5]. Moreover, the method proposed in [2] gives an even better value of CCR%, but it has to be pointed out that a further step of image processing is required if the procedure in [2] is used with respect to the proposed one. Thus, the method reported in [2] seems to be more time-consuming.

4 Conclusions

In this paper, a contribution for improving the detection of diabetic damages has been proposed by synthesizing a Sparsely-Connected Neurofuzzy Network for retinal image processing. A Hopfield-like neurofuzzy subnet firstly highlights vague pale

regions in fundus images of patients with retinopathies. Then, a subsystem constituted by two MLP neural networks has been trained for an optimal thresholding which can minimize pixel classification errors. Contrast-enhanced images have been successively globally segmented via a further neural subnetwork. In resulting bipolar output images diabetic damaged areas are clearly highlighted. Performances of the suggested Sparsely-Connected Neurofuzzy Network prove satisfactory and indicate the possibility of further improvements in the field of ophthalmic diagnostic tools.

Acknowledgements. This work has been supported by Italian Ministry of Scientific Research & Technology, under Grant PRIN no.2004092944_002. The author is sincerely grateful to Dr. Ivo Vulpi of Italian Federation of Red Cross for his important suggestions.

References

1. Hsu, W., Pallawala, P.M.D.S., Li, L.M., Eong, K.A.: The Role of Domain Knowledge in the Detection of Retinal Hard Exudates. In: IEEE Computer Society Conf. on Computer Vision & Pattern Recognition, vol. 2, pp. 246–251 (2001)
2. Walter, T., Klein, J.C., Massin, P., Erginay, A.: A Contribution of Image Processing to the Diagnosis of Diabetic Retinopathy Detection of Exudates in Color Fundus Images of the Human Retina. *IEEE Trans. on Medical Imaging* 21(10), 1236–1243 (2002)
3. Sinthanayothin, C., Kongbunkiat, V., Phoojaruenchanachai, S., Singalavanija, A.: Automated Screening System for Diabetic Retinopathy. In: *Int. Sym. on Image and Signal Processing and Analysis*, pp. 915–920 (2003)
4. Ennett, C.M., Frize, M., Charrette, E.: Improvement and Automation of Artificial Neural Networks to Estimate Medical Outcomes. *Med. Eng. & Phys.* 26, 321–328 (2004)
5. Kavitha, D., Shenbaga, D.S.: Automatic Detection of Optic Disc and Exudates in Retinal Images. In: *IEEE Int. Conf. on Intelligent Sensing and Information Processing*, pp. 501–506 (2005)
6. Gardner, G.G., Keating, D., Williamson, T.H., Elliott, A.T.: Automatic Detection of Diabetic Retinopathy Using an Artificial Neural Network: A Screening Tool. *British J. of Ophthalmology* 80, 940–944 (1996)
7. Osareh, A., Mirmehdi, M., Thomas, B., Markham, R.: Automated Identification of Diabetic Retinal Exudates, in *Digital Colour Images*. *British J. of Ophthalmology* 87(10), 1220–1223 (2003)
8. Swiercz, W., Cios, K.J., Staley, K., Kurgan, L., Accurso, F., Sagel, S.: A New Synaptic Plasticity Rule for Networks of Spiking Neurons. *IEEE Trans. on NN* 17(1), 94–105 (2006)
9. Carnimeo, L.: *Synthesis of Neural Associative Memories for Artificial Vision Systems by Fuzzy Image Segmentations*. Artificial Intelligence Series: Advances in Neural Networks World. World Scient. Eng. Soc. Press, Danvers (2001)
10. Gonzalez, R.C., Woods, R.E.: *Digital Image Processing*, 2nd edn. Prentice Hall, New Jersey (2002)

Dimensionality Reduction Method Applied for Multi-view Face Database Recognition

Hui-Ming Huang^{1,2} and Zhi-Kai Huang¹

¹ Department of Machinery and Dynamic Engineering, Nanchang Institute of Technology, Nanchang, Jiangxi 330099, China

² Institute of Mechanical and Electronic Engineering, Nanchang University, Nanchang Jiangxi 330031, China
huangzhik2001@yahoo.com.cn

Abstract. Faces under varying illumination, pose and non-rigid deformation are empirically thought of as a highly nonlinear manifold in the observation space. How to discover intrinsic low-dimensional manifold is important to characterize meaningful face distributions and classify them using some classifiers. In this paper, we use the Locally Linear Embedding (LLE) algorithm to reduce the dimensionality of face image. The LLE algorithm is the fast dimensionality reduction algorithm that finds local geometry in high dimensional space, and produces a projection to low dimensional space which preserves the original geometry. So, we use the Locally Linear Embedding (LLE) algorithm to reduce the dimensionality of face image for face recognition. Both frontal head images and rotated head images are investigated. Experiments on The UMIST Face Database that is a multi-view database show that the advantages of our proposed approach.

1 Introduction

Machine perception of human faces, or human face image analysis, aims to develop computer algorithms to process, analyze, and recognize the images of human faces. It can be traced back to the 1970s, when Kelly and Kanade started research on automatic machine recognition of faces [1] [2]. Face recognition is a part of a wide area of pattern recognition technology. Face recognition has been a very popular research topic in recent years [3]. It covers a wide variety of application domains, including security systems, personal identification, image and human-computer interaction. The process is influenced by several factors such as shape, reflectance, pose, occlusion and illumination, which make it even more difficult. Today there are many well known techniques trying to recognize a face. A lot of face recognition algorithms, along with their modifications, have been developed during the past decades. The most common techniques are including feature based methods, PCA based eigenfaces[4], LDA based fisherfaces[5][6], ICA[7], Gabor wavelet based methods [8], neural networks and hidden Markov models for face recognition[9][10][11]. However, few can achieve an absolutely reliable performance. The problem arises due to the difficulty of distinguishing different individuals who have approximately the same facial configuration

and yet contend with wide variations in the appearance of a particular face due to changes in pose, lighting, facial makeup and facial expression.

The existing methods for face image analysis can be roughly classified into two categories: feature based (or model based), and appearance based (or learning based). A feature based method employs a set of low-level features (edge/shape, color, texture, motion, and spatial relations) and certain mid-level features (eyes, nose, mouth, hair, and face contours) to determine if there is a human face presented, or verify the identity of this face. This method often requires some prior knowledge, experience, and observations to design the feature extractors. On the other hand, an appearance based method employs the whole images as the input feature vectors, and then low-dimensional features are extracted by some learning algorithms. The difference between the two categories of methods lies in the way how the features are extracted. In a feature based method, features are designed completely by the algorithm designers; in an appearance based method, features are automatically extracted or learned by some smart algorithms.

While numerous methods have been proposed to detect faces in a single image of intensity or color images, but the majority of research has so far focused on frontal face recognition, there is a sizable body of work on pose invariant face recognition and illumination invariant face recognition. At the same time, face recognition across pose and illumination has received little attention.

At the same time, a lot of experimental results show that feature based or model based methods often can not offer satisfied performance due to their incapability of dealing with variations in scale, pose, lighting, and shape. So, most face recognition systems using linear method are bound to ignore subtleties of manifolds such as concavities and protrusions, and this is a bottleneck for achieving highly accurate recognition.

In recent several years, nonlinear dimensionality reduction (NLDR) methods have attracted great attentions due to their capability to deal with nonlinear and curved data.

Locally linear embedding [12] is a flattening method designed for preserving the local structure of the data, and addressing the problem of nonlinear dimensionality reduction. The mapping is optimized to preserve the local configurations of nearest neighbors, while assuming a local linear dependence between them. The 'neighborhood' definitions of each point is set by the user, and may include all points which distances from a given point is smaller than a certain value, a fixed number of closest points, or any other reasonable neighborhood definition.

While there are many impressive results about how to mine the intrinsic invariants of face manifold, manifold learning on face recognition has fewer reports. A possible explanation is that the practical face data include a large number of intrinsic invariant and have high curvature both in the observation space and in the embedded space, and meanwhile the effectiveness of currently manifold learning methods strongly depend on the selection of neighbor parameters.

To address the problem, we present LLE for recovering the intrinsic low-dimensional space embedded face manifold in the observation space. In this paper, we use the LLE algorithm to reduce the dimensionality of face recognize. The LLE algorithm is the fast dimensionality reduction algorithm that finds local geometry in high dimensional space, and produces a projection to low dimensional space which preserves the original geometry. Experiments carried out on UMIST Face Database databases show that advantages of our proposed method. The implementation has been done exclusively using

Matlab and its Image Processing Toolbox. Finally we do some testing on the FERET face database to see how good our implementation is. Both frontal head images and rotated head images are investigated.

2 Algorithm Description and Classifier

2.1 LLE Algorithm Description

The main principle of LLE algorithm is to preserve local neighborhood relation of data in both the embedding space and the intrinsic one. Each sample in the observation space is a linearly weighted average of its neighbors [13]. The algorithm is described as follows:

Given N real-valued vectors \bar{X}_i , each of dimensionality D the algorithm represents each data point (vector) by linear coefficients that reconstruct each data point from its neighbors. It is done by identifying K nearest neighbors per data point as measured by Euclidian distance. (In this stage different metrics and more sophisticated rules can be used). Reconstruction error is measured by the cost function:

$$\mathcal{E}(W) = \sum_i \left| \bar{X}_i - \sum_j w_{ij} \bar{X}_j \right|^2 \quad (1)$$

The weights w_{ij} summarize the contribution of the j -th data point to the i -th reconstruction. To compute the weights we should minimize the cost function subject to two constraints: each data point is reconstructed from its neighbors only (enforcing $w_{ij} = 0$ if \bar{X}_j does not belong to this set), the rows of the weights matrix sum to one ($\sum_j w_{ij} = 1$). The optimal weights found by solving a least squares problem.

In the final step of the algorithm, each high dimensional data point \bar{X}_i is mapped to a low dimensional vector \bar{Y}_i . This is done by choosing d -dimensional coordinates \bar{Y}_i to minimize the embedding cost function:

$$\Phi(Y) = \sum_i \left| \bar{Y}_i - \sum_j w_{ij} \bar{Y}_j \right|^2 \quad (2)$$

The cost function is based on locally linear reconstruction errors (like the previous one). Here we fix the weights w_{ij} while optimizing the coordinates \bar{Y}_i . The above cost function can be minimized by solving a sparse $N \times N$ eigenvector problem. The optimal solution is the smallest eigenvectors of matrix $(I - W)^T (I - W)$. We should choose bottom d non-zero eigenvectors to provide an ordered set of orthogonal coordinates centered on the origin. So we need to compute the bottom $d + 1$ eigenvectors of the matrix and discard the smallest eigenvector.

Because the algorithm has only one free parameter that is the number of neighbors per data point, K . Once neighbors are chosen, the optimal weights and coordinates in the low dimensional space are computed by standard methods in linear algebra.

Locally Linear Embedding Algorithm Pseudo code as follow:

Input X : D by N matrix consisting of N data items in D dimensions.

Output Y : d by N matrix consisting of $d < D$ dimensional embedding coordinates for the input points.

1. Find neighbors in X space.

for $\forall i = 1:N$

 compute the distance from X to every other point X

 find the K smallest distances

 assign the corresponding points to be neighbors of X

end

2. Solve for reconstruction weight W .

for $\forall i = 1:N$

 create matrix Z consisting of all neighbors of X

 subtract X_i from every column of Z

 compute the local covariance $C = Z^T Z$

 solve linear system $C * w = 1$ for w

 set $W_{ij} = 0$ if X_j is not a neighbor of X

 set the remaining elements in the i -th row of W equal to $w / \text{sum}(w)$

end

3. Computing embedding coordinates Y using weights W .

 create sparse matrix $M = (I - W)^T (I - W)$

 find bottom d eigenvectors of M

 (corresponding to the $d+1$ smallest eigenvalues)

set the q -th row of Y to be the $q+1$ smallest eigenvector

(discard the bottom eigenvector $[1,1,1,\dots]$ with Eigen value is zero)

The Principle scheme of the Locally Linear Embedding (LLE) algorithm has been shown in figure 1.

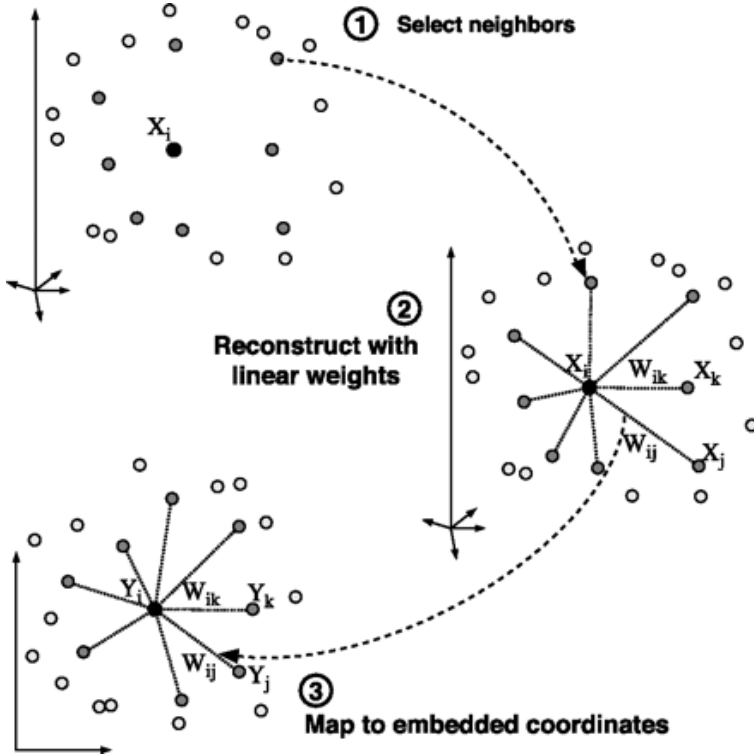


Fig. 1. Principle scheme of the Locally Linear Embedding algorithm

2.2 K-Nearest Neighbor Classification

In classification, k-nearest neighbor classification principle was used. At classification of unknown object there is a given number (k) vectorally the nearest to it in space of attributes of other objects (the nearest neighbors) with already known belonging to recognized classes. The decision on reference of unknown object to this or that diagnostic class is accepted by the analysis of information on this known belonging of its nearest neighbors, for example, with the help of simple calculation of voices. Originally the method of the k-nearest neighbors was considered as a nonparametric method calculates relations of plausibility. At use of a method of the k -nearest neighbors for

recognition of images the researcher should solve a complex problem of choice the metrics for definition affinity of diagnosed objects. This problem in conditions of high dimension of space of attributes extremely becomes aggravated owing to sufficient labor input of the given method which becomes significant even for high-efficiency computers.

Therefore here the same as and in a method of comparison with the prototype, it is necessary to solve a creative task of analysis multivariate structure of experimental data for minimization of number of the objects representing diagnostic classes.

Necessity of reduction number of objects for training sample (diagnostic precedents) is lack of the given method as reduces imposing appearance of training sample.

3 Experimental Data and Results

UMIST face database is a multi-view database, consisting of 575 images of 20 people with varied poses. The images of each subject cover a range of poses from right profile (-90 degree) to frontal (0 degree) [15]. Examples of the UMIST database are shown in Figure 3. The mainly difficulty of UMIST database is that face data in the observation space may have higher curvature and stronger nonlinearity in multiple views than in frontal views. From the aspect of computer vision, meanwhile, "the variations between the images of the same face due to illumination and viewing direction are almost always larger than image variations due to change in face identity"[15]. This makes multi-view face recognition a great challenge.



Fig. 2. Summarizes Sample Images of a Typical Subset in the Database

As we remember, LLE algorithm has only one free parameter – the number of neighbors per data point, K . Once neighbors are chosen, the optimal weights and coordinates in the low dimensional space are computed by standard methods in linear algebra.

So before testing LLE performances we should decide which number of neighbors to use for current type of data. To do this we measured classification error as function of neighbor's number for each type of data (reducing the dimensions to 4). And latter the performances' testing was done with value, which gives best results for this type of data.

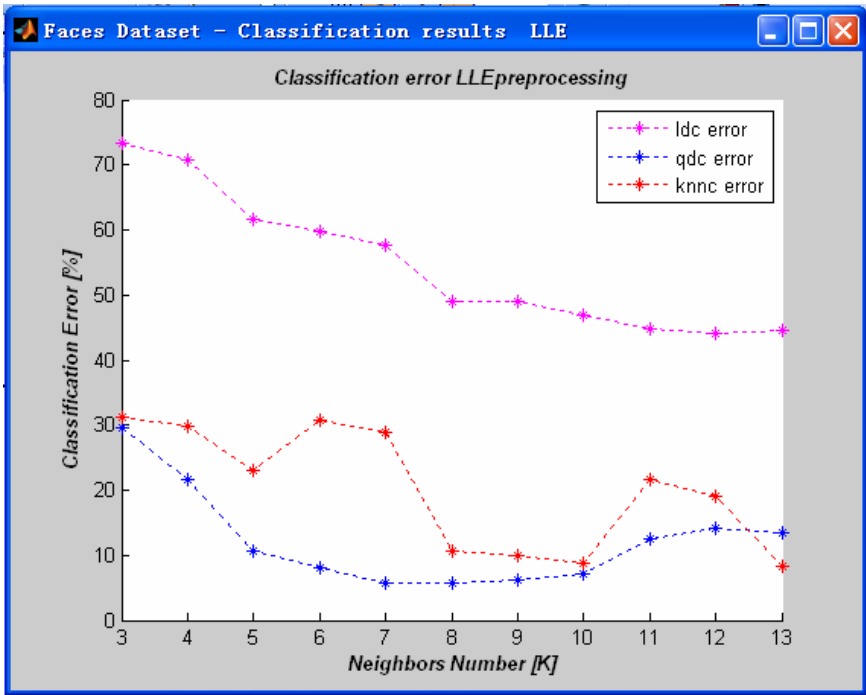


Fig. 3. Classification Results Used Three Classification Rules

For getting more righteous estimation three classification algorithms were used: LDC (Linear Discriminant Classifier, LDC that finds the line that best separates the points), QDC (Quadratic Discriminant Classifier) and KNNC. That way we also might perform another check for each one from the tested algorithms: in combination with which one from the used classifiers it works better. For performing of the classification we used PRTools classification toolbox [14]. The toolbox also supplies some useful functionality like generating datasets, splitting existing datasets to learning dataset and test one, displaying the data features map and classified data scattered plots and more.

From the above graph we can see that for *ldc* and *qdc* the minimum error is for 10 neighbors and for *knnc* the minimum error is for 10 neighbors. The classification results are presented in Figure 2; the graphs show the classification rate (y axis) with respect to the dimension reduction (x axis). The better results are obtained in 40 dimensions and computing times are large. On the other hand, LDA increases also recognition results according to all classification rules. LDA gives a very discriminate representation in ($k > 1$) dimensions (here $k = 9$) which allows to use efficiently the rule. Consequently, classification results are good and computing times are small.

4 Conclusions and Future Work

In this paper, we use the Locally Linear Embedding (LLE) algorithm to reduce the dimensionality of face image. The LLE algorithm is the fast dimensionality reduction

algorithm that finds local geometry in high dimensional space, and produces a projection to low dimensional space which preserves the original geometry. We use the Locally Linear Embedding (LLE) algorithm to reduce the dimensionality of face image for face recognition. Both frontal head images and rotated head images are investigated. Experiments on The UMIST Face Database that is a multi-view database show that the advantages of our proposed approach. Results indicate that the proposed methodology is able to achieve a certain extent performance. All current face recognition algorithms fail under the vastly varying conditions under which humans need to and are able to identify other people. As mentioned at the end of the introduction, the environments where face images are taken are far from being controlled. Usually, the background is complex, person happens to be anywhere in the image, can be of an arbitrary rotation and size. In addition, generally images of the same face may look very different due to differences in both illumination direction and intensity. Since the performance of the overall scheme would considerably degrade if input images do not fulfill certain constraints, it is wise to preprocess the images in a manner to maximize to performance. So, the face image that used LLE preprocessing is carried out to increase its robustness against variations in poses and illumination level. Face recognition rate can benefit from preprocessing the data such that the LLE step would be empowered. In the future, person recognition systems will need to recognize people in real-time and in much less constrained situations. We will study other efficient detection algorithms and will integrate global features to further improve the recognition performance of this system.

References

1. Kanade, T.: *Picture Processing by Computer Complex and Recognition of Human Faces*, Kyoto University, Ph.D. thesis edition (1973)
2. Kelly, M.D.: *Visual Identification of People by Computer*. Tech. Report AI-130, Stanford AI Project, Stanford, CA, USA
3. Zhao, W., Chellappa, R., Phillips, P.J., Rosenfeld, A.: *Face Recognition: A Literature Survey*. *ACM Computing Surveys* 34(4), 399–458 (2003)
4. Kirby, M., Sirovich, L.: *Application of the Karhunen-Loeve Procedure for the Characterization of Human Faces*. *IEEE Trans. Pattern Analysis and Machine Intelligence* 12(1), 103–108 (1990)
5. Swets, D., Weng, J.: *Using Discriminant Eigenfeatures for Image Retrieval*. *IEEE Trans. on PAMI* 18(8), 831–836 (1996)
6. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: *Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection*. *IEEE Trans. Pattern Analysis and Machine Intelligence* 19(7), 711–720 (1997)
7. Bartlett, M.S., Movellan, J.R., Sejnowski, T.J.: *Face Recognition by Independent Component Analysis*. *IEEE Transaction on Neural Networks* 13, 1450–1464 (2002)
8. Kepenekci, B.: *Face Recognition Using Gabor Wavelet Transform*. MSc. Thesis, METU (September 2001)
9. Liu, C.J., Wechsler, H.: *Gabor Feature Based Classification Using the Enhanced Fisher Linear Discriminant Model for Face Recognition*. *IEEE Trans. on Image Processing* 11(4), 467–476 (2002)

10. Huang, D.S., Zhao, W.B.: Determining the Centers of Radial Basis Probabilities Neural Networks by Recursive Orthogonal Least Square Algorithms. *Applied Mathematics and Computation* 162(1), 461–473 (2005)
11. Huang, D.S.: *Systematic Theory of Neural Networks for Pattern Recognition*. Publishing House of Electronic Industry of China, Beijing (1996)
12. Phillips, P.J., Moon, H., Rizvi, S.A., Rauss, P.J.: The FERET Evaluation Methodology for Face Recognition Algorithms. *IEEE Trans. on PAMI* 22(10), 1090–1104 (2000)
13. Tenenbaum, J.B., Silva, V.D., Landford, J.C.: A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science* 290, 2319–2323 (2000)
14. PRTools Classification Toolbox,
<http://www.prtools.org/cgi-bin/twiki/bin/view>
15. Graham, D.B., Allinson, N.M.: Characterizing Virtual Eigensignatures for General Purpose Face Recognition. In: Wechsler, H., Phillips, P.J., Bruce, V., Fogelman-Soulie, F., Huang, T.S. (eds.) *Face Recognition: From Theory to Applications*. NATO ASI Series F, Computer and Systems Sciences, pp. 446–456 (1998)

Experimental System to Support Real-Time Driving Pattern Recognition

Vincenzo Di Lecce* and Marco Calabrese

Politecnico di Bari - DIASS
Viale del Turismo 8, 74100, Taranto, Italy
{v.dilecce, m.calabrese}@aeflab.net

Abstract. This work proposes an advanced driving information system that, using the acceleration signature provided by low cost sensors and a GPS receiver, infers information on the driving behaviour. The proposed system uses pattern matching to identify and classify driving styles. Sensor data are quantified in terms of fuzzy concepts on the driving style. The GPS positioning datum is used to recognize trajectory (rectilinear, curving) while the acceleration signature is bounded within the detected trajectory. Rules of inference are applied to the combination of the sensor outputs. The system is real-time and it is based on a low-cost embedded lightweight architecture which has been presented in a previous work.

Keywords: driving patterns, acceleration signature, fuzzy inference system.

1 Introduction

For many years there has been a widespread consensus on the benefits of Advanced Traveller Information Systems (ATIS) in enhancing personal mobility, safety and the productivity of transportation. The primary services of ATIS include pre-trip and/or en route traveler information concerning traffic conditions, route guidance, and "yellow page-type" information related to traveling as well as entertainment, dining and other services [9]. The basic ATIS architecture consists of a network of cameras connected to a processing center, thus giving the system user a real-time feedback on the whole traffic conditions and many other important pieces of information. This however does not solve all problems like safe driving which strictly depends on driver's behaviour along with his/her attitude to respect driving laws. Furthermore, the ATIS approach needs to be endowed with complex object recognition software to produce statistics on traffic flows or detect illegal driving behaviours. An alternative approach (that can be interesting for car insurance companies as well as public authorities interested in public safety) can be based on directly monitoring of driving behaviours using the vehicle as a probe. This can be accomplished by using low-cost devices that provide real-time driving information. This work presents such a system that, using a light-weight embedded low-cost architecture, is able to support real-time monitoring

* Corresponding author.

of driving behaviours. Up to the authors' knowledge this approach is quite novel in the context of driving information systems. The system is composed of a processing unit equipped with a GPS receiver and two biaxial accelerometers. Information coming from the two sources is fused within the processing unit, hereinafter technically referred as Device Server Unit (DSU), a hand-held device which requires very little space and no particular skills to be correctly mounted on board. The rest of the paper is divided as follows: Section II illustrates related work on driving pattern recognition, Section III describes the proposed architecture, Section IV specifically accounts for pattern analysis and experiments. Finally, conclusions are drawn in Section V.

2 Related Work

Driving pattern recognition is an interdisciplinary topic that has been widely addressed under many points of view. One of them is represented by research about autonomous vehicle driving. In this field, three main approaches are commonly considered: neural network, explicit modeling and pattern matching [6]. Each of them handles different characteristics with respect to the way knowledge domain is structured and expressed. Neural networks and explicit modeling represent two supervised ways to teach a system how to follow a certain objective: in the first case an appropriate set of training samples is required, in the second case many parameters need to be tuned in order to get valid results. It is also possible to have a mixed approach. In [13] the overtaking task is considered and modeled in system theoretic terms; model results are then used to teach a neural network. An alternative approach is pattern matching. It basically consists in having a pre-built knowledge on some characteristics of road marks and therefore weighing the probability that the real-time system state corresponds to one of those predefined road marks. Pattern matching encompasses the problem of defining an appropriate metric to estimate discordance between actual and recorded knowledge. However, in the case of driving behaviour, the cardinality of pattern set is quite small i.e. the alphabet of possible driving patterns is very limited.

As deeply investigated in the literature, the driving style can be affected by a number of different and independent reasons. Some researchers [11, 4] even identify driving behaviour signals as biometric information. They use force on brake and force on accelerator for this scope. These elements appear to be very sensitive to human subjectivity and they influence car-driving attitudes also in the presence of physical constraints like car following or maintaining lane. [5] for example refer to car-following behavior in Intelligent Driver Assistance Systems (ISAS). The authors assess the possibility that the current state between the lead and the following vehicle determines entirely the future state of the following vehicle, with no dependence on the past sequences of car motions that produced the current state. Other approaches [10] merge the two parameters of car-following and car-pedal use for enabling ISAS with customized assistance for drivers. Updated ISAS take also care of the ride comfort and there can be found some research about it, in particular [12]. Apart from car-following, other specific driving situations can be taken into account such as when approaching to intersection [12], when stopping or overtaking and so on. It is noteworthy however that, in many of these works, acceleration signature is taken in high consideration (see, for example [7]).

The DSU represents the core system architecture, interposing between the sensor devices and the application server. By modelling the signal path from the sensors to the application server, it is clear that the DSU is the core unit. Two possible configurations can be implemented depending on both the system user's needs and performance requirements: stand-alone and client-server. In the stand-alone configuration DSU interfaces directly with the driver. In this case the system represents an add-on to commonly used GPS car navigation systems. In the client-server configuration sensor data is locally pre-processed and successively sent to a remote application server (i.e. for data mining purposes). Remotely monitoring real-time driving information can be in fact useful for a number of applications such as (to cite only a limited number) identifying safe truck driving conditions for hazardous material transportation, supervising correct liquid bulk tanks loading/unloading, monitoring ambulance fleets.

4 Pattern Analysis

This section presents how to use GPS positioning information in combination with the acceleration signature provided by biaxial accelerometer sensors to characterize driving patterns. In more details, GPS provides trajectory information, while accelerometers account for the way a driver, given a certain trajectory, uses brake/acceleration as driving inputs. The two data sources can be combined in a Fuzzy Inference System (FIS) to infer about driving styles. In this paper we focus on both the acceleration and GPS patterns, providing an overview of the whole framework comprising the fuzzy logic-based part.

4.1 Accelerometers

Biaxial accelerometers are inertial sensors for measuring acceleration along two orthogonal input axes. In our tests they have been mounted in a strap-down configuration with the two axes positioned one along the moving direction and the other on the left to it. Because they sense changes to inertia they respond positively to deceleration and negatively to accelerations along their input axes. They generally suffer from bias deviation from V_{cc} and, depending on their cost and manufacturing, they show a mean distance from the true value of 3-4% for very cheap ones.

From the above considerations, given the x-y plain representing the two input axes we can define seven fuzzy regions representing different acceleration conditions:

- **Uncertainty Area:** it can be assumed to be a circular region centered in the unbiased origin. It accounts for instrument inaccuracy or other undistinguishable states such as vehicle stopped or moving with a constant speed.
- **Acceleration Area:** it represents the region where acceleration along the moving direction is sensed. It can be essentially imputed to the driver push of the gas pedal, but it can be also related to a declining altitude
- **Deceleration Area:** it is the opposite of the previous one
- **Remaining areas of the four quadrants:** which account for curve entering or leaving

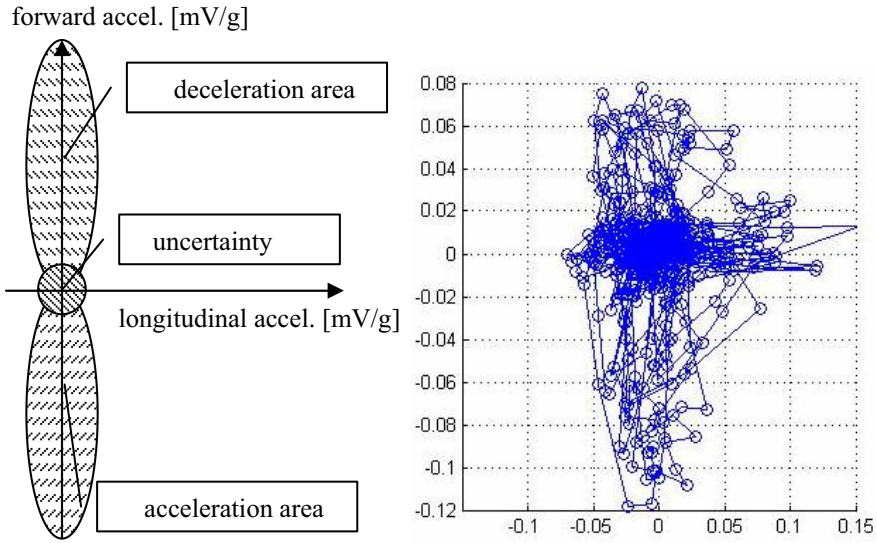


Fig. 2. (left) Pictorial representation of the areas defining different acceleration situations and (right) plot of real data acquisition. The unit measure is [mV/g] because the sensor output is a voltage ($1g = 312mV$ for the sensor used). Data have been detrended. The offset value was about 2.5 V.

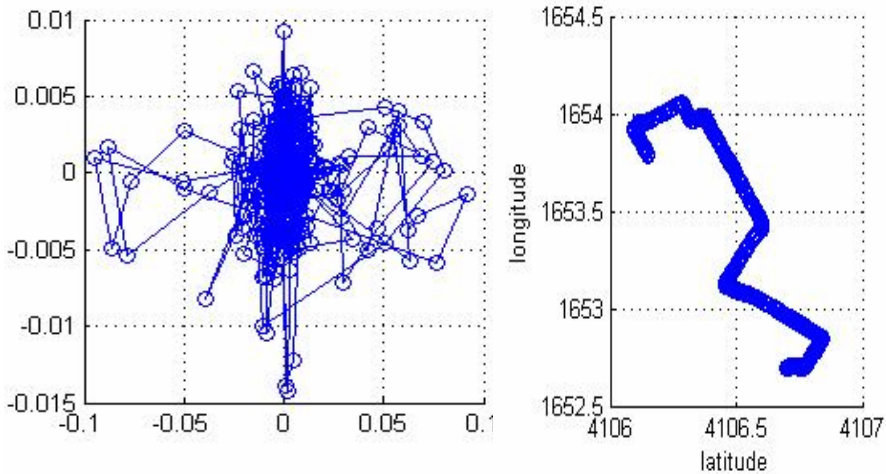


Fig. 3. Accelerometer and GPS data of the same test acquisition in comparison. Accelerometer signature (left side) shows more density patterns in entering/leaving left-handed curve. This assumption is confirmed by the nature of the actual trajectory (right side). Vehicle starts in the lower-right part and progressively moves to the upper-left thus following (principally) a westward direction.

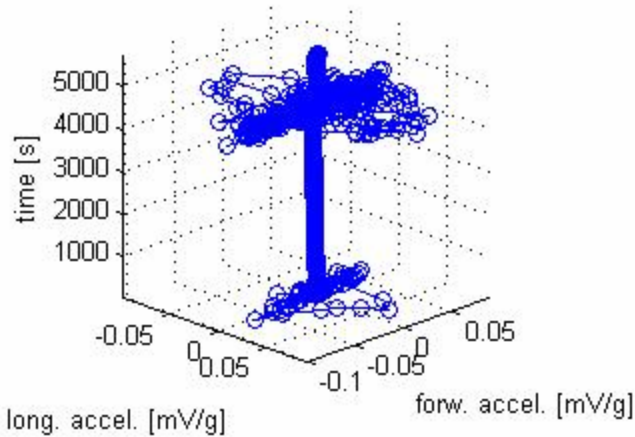


Fig. 4. 3D representation of the data presented in Figure 3. Z axis accounts for the time of the acquisition. From this perspective it is clear that the vehicle, after some short manoeuvres has stopped for nearly an hour before moving again.

The boundaries of the classified areas should be intended as fuzzy. This is the reason why a good way to handle them analytically is using a fuzzy approach. Fig. 2 depicts the comparison between the proposed classification and a real 10-minute acquisition. A plot of accelerometer data considering time passing is depicted in Fig. 4.

4.2 GPS

A GPS receiver provides information in terms of latitude, longitude and altitude at a given frequency (usually 1 second for most GPS receivers). Then, it is easy to reckon the mean speed as the vector difference between two consecutive GPS positioning acquisitions. Therefore, two differential variables can be extracted from the GPS receiver: the angular variation and the mean speed. Similarly to the accelerometer case, these values can be handled analytically using a fuzzy approach. It is noteworthy that data provided by GPS are strictly correlated to the accelerometer output. This means that GPS should “confirm” inferences deriving from accelerometer sensor only. To have an evidence of this, Figure 3 displays a test comparison.

4.3 Fuzzy Inference System

A FIS can be considered as an inference system that maps, by means of combination rules, input to output using fuzzy logic. A huge literature exists on this topic ([14], [2], [8] to cite only a few) and a wide variety of both academic and industrial applications is available today. Up to the authors’ knowledge however, little contribution can be found about the topic discussed in this work.

Using the four inputs provided by the accelerometer sensor (forward and longitudinal acceleration) and the GPS receiver (angular variation and mean speed) a FIS can be

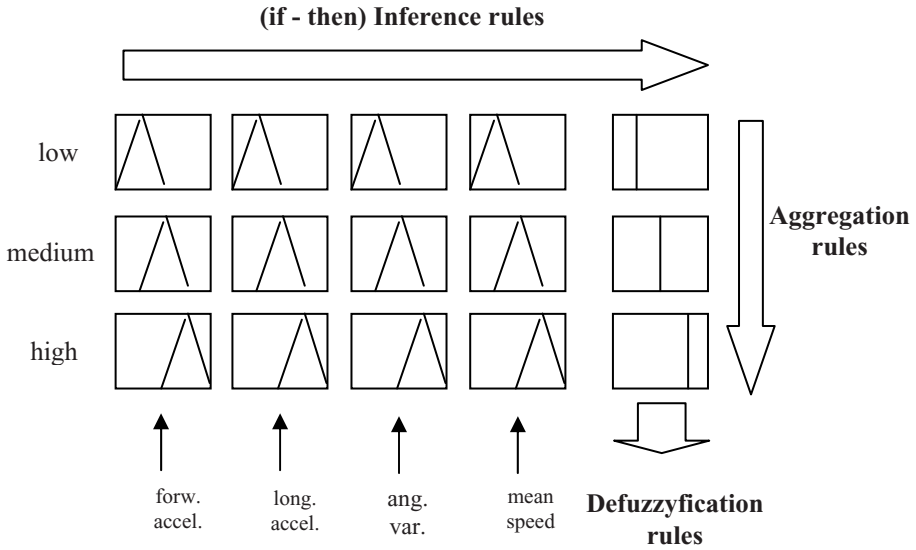


Fig. 5. Logical representation of a FIS for supporting driving information discovery

designed, according to the level of detail that one wishes to have. In a preliminary approach, three triangular membership functions representing “low”, “medium” and “high” fuzzy concept values have been used in a Sugeno-like FIS architecture (Figure 5).

5 Conclusion

An experimental advanced driving information system that uses the acceleration signature provided by low cost sensors and a GPS receiver to infer on driving behaviour has been presented. Although this research is at an early stage, a prototypal architecture has been implemented for testing purposes. First results seem promising since they capture the “alphabet” of meaningful driving patterns in an efficient way. The architecture is robust and flexible enough to be adapted to various configurations depending on the user need and performance requirements. Further research will be primarily focused on better defining driving semantics, improving pattern detection algorithms and finally testing different inference rules with respect to the specific applications.

References

1. Akamatsu, M.: Measuring Driving Behavior. In: Annual conference of the Society of Instruments and Control Engineers, SICE 2002 (2002)
2. Bezdec, J.C.: Pattern Recognition with Fuzzy Objective Function. Plenum Press, New York (1981)

3. Di Lecce, V., Amato, A., Calabrese, M.: GPS-aided Lightweight Architecture to Support Multi-sensor Data Synchronization. In: I2MTC 2008, Proc. of IEEE International Instrumentation and Measurement Technology Conference, Vancouver, Canada, May 12-15 (to appear, 2008)
4. Igarashif, K., Miyajimar, C., Itout, K., Takedai, K., Itakurat, F., Abut, H.: Observation of Real Driving Behavior in Car-Following: Preliminary Results, Biometric Identification Using Driving Behavioral Signals. In: IEEE International Conference on Multimedia and Expo (ICME), Proc. of ICME, pp. 64–68 (2004)
5. Kim, T., Lovell, D.J.: Observation of Real Driving Behavior in Car-Following: Preliminary Results. In: IEEE 61st Vehicular Technology Conference. VTC 2005 (2005)
6. Krodel, M., Kuhuert, K.D.: Pattern Matching as the Nucleus for Either Autonomous Driving or Driver Assistance Systems. In: Intelligent Vehicle Symposium, IEEE, vol. 1, pp. 135–140 (2002)
7. Li, Y., Donald, M.M.: Link Travel Time Estimation Using Single GPS Equipped Probe Vehicle. In: ITSC-3rd Intl. Conference on Intelligent Transortation Systems (2002)
8. Mamdani, E.H., Assilian, S.: An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller. *International Journal of Human-Computer Studies* 51(2), 135–147 (1999)
9. Mouskos, K.C., Greenfeld, J., Pignataro, L.J.: Toward a Multi-Modal Advanced Traveler Information System. *NJIT Research*, vol. 4 (1996)
10. Miyajima, C., Nishiwaki, Y., Ozawa, K., Toshihiro, K., Itou, K., Takeda, K., Itakura, K.: Driver Modeling Based on Driving Behavior and Its Evaluation in Driver Identification 95(2) (2007)
11. Wakita, T., Ozawa, K., Miyajima, C., Itou, K., Takeda, K., Itakura, K.: Driver Identification Using Driving Behavior Signals. *IEICE - Transactions on Information and Systems archive E89-D(3)*, 1188–1194 (2006)
12. Wang, F., Ma, N., Inooka, H.: A Driver Assistant System for Improvement of Passenger Ride Discomfort through Modification of Driver Behaviour. In: ADAS 2001: International Conference on Advanced Driver Assistance System (2001)
13. Wewerinke, P.H.: Model Analysis of Adaptive Car Driving Behavior. In: IEEE International Conference on Systems, Man, and Cybernetics, vol. 4, pp. 2558–2563 (1996)
14. Zadeh, L.: Knowledge Representation in Fuzzy Logic. *IEEE Transactions on Knowledge and Data Engineering* 1, 89–100 (1989)

Extending Hough Transform to a Points' Cloud for 3D-Face Nose-Tip Detection

Vitoantonio Bevilacqua^{1,2}, Pasquale Casorio¹, and Giuseppe Mastronardi^{1,2}

¹ Dipartimento di Elettrotecnica ed Elettronica Politecnico di Bari
Via Orabona, 4 70125 - Bari - Italy
bevilacqua@poliba.it

² e.B.I.S. s.r.l. (electronic Business in Security), Spin-Off of Polytechnic of Bari,
Str. Prov. per Casamassima Km. 3-70010 Valenzano (BA)-Italy

Abstract. This paper describes an extension of Generalized Hough Transform (GHT) to 3D point cloud for biometric applications. We focus on the possibility to applying this new GHT on a dataset representing point clouds of 3DFaces in order to obtain a nose-tip detection system.

1 Introduction

In Computer Vision object or shape detection in 2D/3D images is very hard to solve because shapes can be subject to translations, can change by color, can be subject to scale and orientation, can endure occlusions and moreover data acquisition can introduce high levels of noise. One of the more effective solutions for shape detection is the Hough Transform. Formulated for the first time in early '60s, it originally was able to recognize shapes that had analytical description such as straight lines, circles and ellipses in 2D intensity images.

In 1981 Ballard [1] proposed an extension defined Generalized Hough Transform (GHT) for generic shape detection by using the R-Table, a table that describes the shape to search respect to a reference point that could represent the center of the pattern. Many efforts have been done in order to try to extend the GHT in three-dimensional images. Khoshelham [2] proposed an extension of Ballard GHT for three-dimensional images constituted by point clouds obtained by means of laser-scanner acquisitions, for generic applications.

This work adapts Khoshelham GHT in an attempt to apply it on 3D-Face shaded model for nose-tip detection. In this way it is possible to create an automatic repere's points detection system with the purpose of obtaining a biometric system for AFR (Automatic Face Recognition) using 3DFace templates.

The research was lead on a database of 3D-faces in ASE format, the GavaDB, given by the GAVAB research group of computer science department at the University of King Juan Carlos in Madrid.

2 An Overview of the Standard and Generalized Hough Transform

2.1 The Standard Hough Transform

The idea of Hough transform for detecting straight lines in 2D intensity images was first introduced by Hough in 1962. It uses the Hesse normal form to represent a straight line:

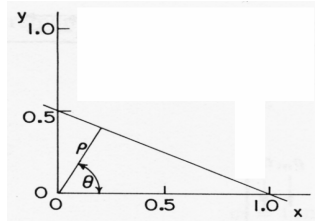


Fig. 1. The straight line and its Hesse normal form parameters

$$\rho = X \cos \theta + Y \sin \theta \tag{1}$$

The parameters ρ and θ represent the distance of the application point of the normal vector from the origin of XY reference system, and the angle that it forms with the X-axis. In this way it is possible to consider a transformation from the plan of the image (on which the shape it is represented) to the space of the parameters:

- A straight line in XY space corresponds to a point (ρ', θ') in the parameter space.
- Each pixel in the image corresponds to a curve representing all the parameters of the lines of a bundle of straight that have that pixel as intersection point .
- Fixing two pixels, they identify one and one single straight. Such situation is represented in the space of the parameters by means of an intersection of the curves correspondents to the same points.

The parameter space is realized in the form of a discrete accumulator array $H[\theta][\rho]$, consisting of a number of bins that receive votes from edge pixels in the image space.

The algorithm will be:

- Bound ρ and θ in order to obtain
- $0 \leq \rho \leq \text{Max}$, $0 \leq \theta \leq 2\pi$, where $\text{Max} = 0,5 \cdot \sqrt{(\text{NPixelRow}^2 + \text{NPixelColumn}^2)}$
- Choose the level of accumulator array quantization basing on the desired accuracy.
- Annul the H array.
- For each pixel P having coordinates (x, y)
 - for θ_n in $[0, 2\pi]$ with step $d\theta$
 1. Evaluate $\rho(n) = x \cdot \cos(\theta_n) + y \cdot \sin(\theta_n)$
 2. Obtain the index m that corresponds to $\rho(n)$
 3. Increase $H(m, n)$ of one vote.
 - End
- End

Find straight lines parameters corresponding to the accumulator bins that received the most numbers of votes.

2.2 The GHT Proposed by Ballard [1]

The standard Hough transform is only able to detect shapes described by parametric equations. However, in real applications the pattern to detect can be so complex than it could not be represented by any analytic description. In 1981, D. H. Ballard proposed a HT extension for generic shape detection.

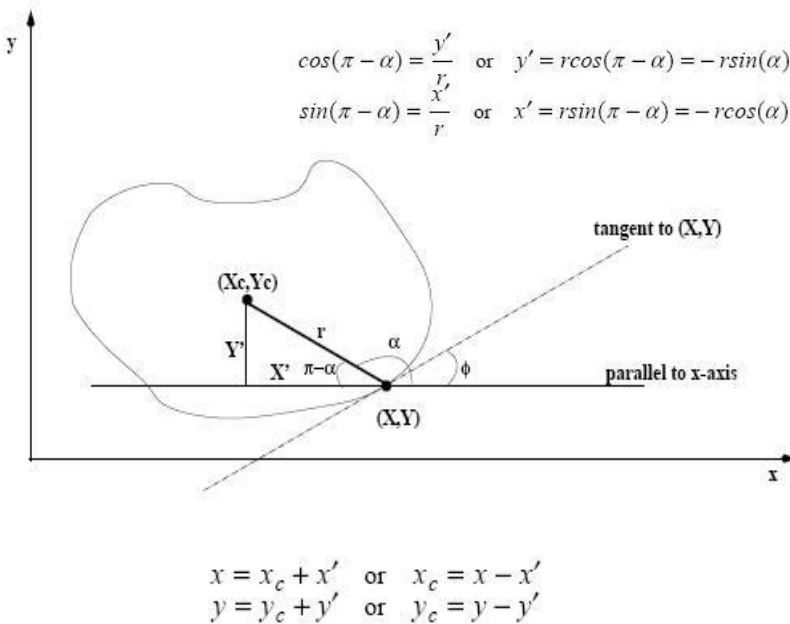


Fig. 2. Example of pattern and equations involved in Ballard’s method

We can arbitrary choose a reference point (X_c, Y_c) representing the centre of the pattern, and several points (X, Y) on the shape edge. The angle Φ that the tangent line forms at the edge point with the X axis, represents towards of route of the edge. Each edge point can be described by an ordered couple such as (r, α) like figure 2 suggests. Now, if a (X, Y) pixel image is an edge point of a pattern centered in (X_c, Y_c) , it satisfies the equation (2):

$$\begin{cases} X_c = X + r \cos \alpha \\ Y_c = Y + r \sin \alpha \end{cases} \quad (2)$$

All the couples (r, α) can be written in a table, called R-Table, sorting them in order to each row contains all the couple which have the same Φ value.

$$\begin{aligned}
 \phi_1 &: (r_1^1, \alpha_1^1), (r_2^1, \alpha_2^1), \dots \\
 \phi_2 &: (r_1^2, \alpha_1^2), (r_2^2, \alpha_2^2), \dots \\
 &\dots \\
 \phi_n &: (r_1^n, \alpha_1^n), (r_2^n, \alpha_2^n), \dots
 \end{aligned}$$

Fig. 3. Example of R-Table

The GHT algorithm is:

- The parameters space that is represented by accumulator array H, corresponds to XY coordinates of the reference point (pattern centre). Quantization level is chosen in order to obtain the desired accuracy level. Initially each accumulator bin is null.
- For each image's pixel (Xp, Yp) to examine evaluate Φ angle and research the R-Table row indexed by its quantization.
- For each couple (r, α) found in the R-Table , calculate
 - $X_c = X_p + r \cos \alpha$
 - $Y_c = Y_p + r \sin \alpha$
 - Increase H[Xc][Yc] of one vote.
- Find the most voted bins. They represent the centre points where shape occurrences are localized into the image.
- If needed, using the centers obtained in previous step to rebuild the pattern as visual reply.

3 The GHT for 3D-Point Clouds by Khouroush Khoshelham [2]

The Generalized Hough Transform is larger employed because of :

- It transforms shape detection problem (very hard to solve) into a maximum analysis problem, that is simpler and simpler than the original one.
- It is robust to partial or slightly deformed shapes (i.e., robust to recognition under occlusion).
- It is robust to the presence of additional structures in the image (i.e., other lines, curves, etc.).
- It is tolerant to noise.
- It can find multiple occurrences of a shape during the same processing pass.
- However, these advantages are not “free of charge”, in fact:
- We need to build a separate *R-table* for each different object.
- It requires a lot of storage and extensive computation (but it is inherently parallelizable!).

If we want to include pattern scale and rotation, it is necessary to add these parameters to the accumulator array, and the computational cost could become too high.

All the HT and GHT have been created for shape detection in two-dimensional images.

One of most interesting challenges consists of obtaining a GHT model to detect patterns in 3D-images, i.e. point clouds, representing objects acquired by means laser scanner. K. Khoshelham proposed a GHT model that we can consider as Ballard’s GHT extension for 3D-images.

Considering an arbitrary three-dimensional pattern (figures 4 and 5), the edge is by a generic surface. Edge vertex normal vectors could give information about towards of route of the surface such as Φ angle does in Ballard’s GHT.

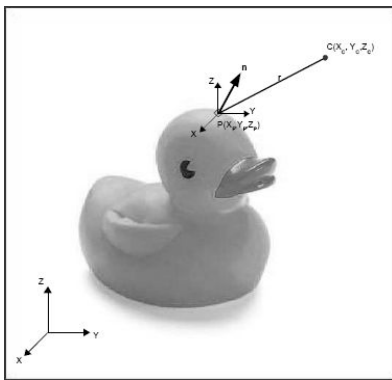


Fig. 4. Example of a generic pattern

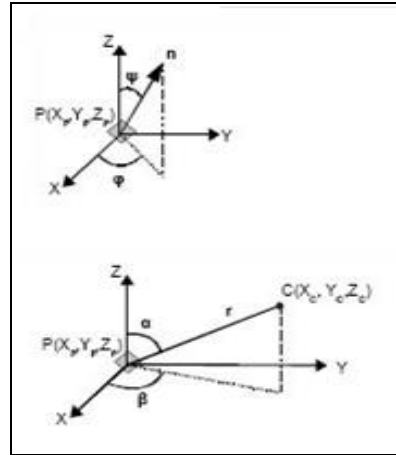


Fig. 5. Parameters involved in R-Table build process

The algorithm of R-Table creation can therefore be modified:

- Examine all the pattern vertexes choosing a centre as reference point. This one will be randomly chosen. A method could consist of evaluate the mean values of pattern’s vertexes coordinates.
- Evaluate for each point P of the pattern

$$\begin{aligned}
 a. \quad r &= \sqrt{(X_c - X_p)^2 + (Y_c - Y_p)^2 + (Z_c - Z_p)^2} \\
 b. \quad \alpha &= \cos^{-1} \left(\frac{Z_c - Z_p}{r} \right) \\
 c. \quad \beta &= \cos^{-1} \left(\frac{X_c - X_p}{r \sin \alpha} \right)
 \end{aligned}
 \tag{3}$$

[Both α and β in $[0, \pi]$]

- Evaluate vertex normal vector angles ψ and ϕ , and quantize them in order to obtain the R-Table indexes.
- Insert the tern (r, α, β) into the R-Table shown in Figure 6.

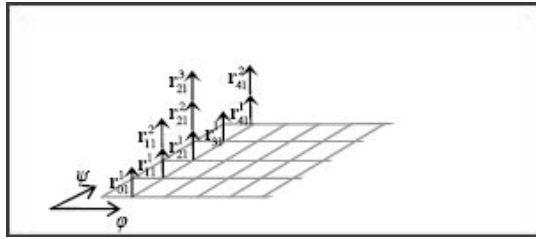


Fig. 6. Example of 2D R-Table

The algorithm consists of these steps:

- Build the accumulator array H and null it. The accumulator represents the X, Y and Z coordinates of the pattern center in the point cloud.
- Read and store in memory the accumulator array H.
- For each point of the cloud to examine:
 - Evaluate its normal vector angles and quantize them in order to obtain the R-Table indexes.
 - For each tern (r, α, β) extracted using previous indexes:
 - Evaluate

$$\begin{cases} X_c = X_p + r \sin \alpha \cos \beta \\ Y_c = Y_p + r \sin \alpha \sin \beta \\ Z_c = Z_p + r \cos \alpha \end{cases} \quad (4)$$

- Increase $H[X'_c][Y'_c][Z'_c]$ of one vote. X'_c, Y'_c and Z'_c represent the quantized values of centre coordinates .
- Find the center coordinates corresponding to the accumulator bins that received the most numbers of votes.
- If needed, using the centers obtained in previous step to rebuild the pattern as visual reply.

If we want to consider geometric transformations such as rotation and scale involving the pattern in the point cloud, the algorithm may be modified. Equation (4) should be changed with (5), express in vector form:

$$\mathbf{c} = \mathbf{p} + s \mathbf{M}_z \mathbf{M}_y \mathbf{M}_x \mathbf{r} \quad (5)$$

Where:

- $\mathbf{c} := (X_c, Y_c, Z_c)^T$
- $\mathbf{p} := (X_p, Y_p, Z_p)^T$
- $\mathbf{r} := (r \sin(\alpha) \cos(\beta), r \sin(\alpha) \sin(\beta), r \cos(\alpha))^T$
- $s :=$ scale factor
- $\mathbf{M}_z, \mathbf{M}_y, \mathbf{M}_x :=$ axis rotation matrixes.

These changes lead the accumulator array to be 7-dimensional array. Three dimensions represent the centre coordinates, one dimension represents the scale factor, and the other three ones represent the rotation along the X, Y and Z axis. So, voting process will generate both rotational and scale factor values by attempts.

However, this solution remains theoretical because of its very high computational costs. Our first 3D-GHT application implements the original algorithm, without scale and rotation, and even in this simpler case the accumulator array uses about 200MB of RAM for memory allocation, to obtain the desired accuracy. The algorithm takes about 10 seconds to detect shapes in point clouds consisted of ten thousands vertex-elements. Our second 3D-GHT application improves previous version implementing pattern scale by means a simplified version of equation 5:

$$c = p + s r \quad (6)$$

A “for-loop” fixes the scale factor before evaluating equation 6. So, it uses an [331]x[331]x[331]x[8] accumulator array. Fourth dimension represents scale factor. With a scale factor’s step of 0.25, the algorithm is able to find 3D-shapes which are in dimensions from 0.25 to 2 times original pattern’s ones.

This algorithm uses about 1GB of RAM and takes less than 20 seconds to detect shape in point clouds consisted of ten thousands vertex-elements.

Concerning time complexity, all the HT algorithms are $O(N)$, where N represents the number of point cloud’s vertexes.

4 The 3D GHT at Work on 3D Faces

The first GHT application (without pattern scale) was tested on a 3D-Face database. Faces have been acquired with different poses. The patterns were spheres with beams of 3.5, 4 or 5 (adimensional values).

Maximum analysis demonstrated how most voted sphere with 3.5 and 5 beams were next to the nose in 80% of examined faces.

This result was used to create an automatic nose-tip detection application. Nose-tip, in fact, is the most important repere point, and it is used in biometrics as parameter in order to study face symmetry or face recognition by means repere points’ triangulation.

The research algorithm can be described by these steps:

1. Research the most voted 3.5 beam sphere. In case of fault, a 5 beam sphere is used. The beams are adimensional values because is impossible extracting from the database information about scale factor involved in data acquisition.
2. Extract face vertexes in proximity of the found sphere, magnifying extracted cloud of a factor of 10.
3. Research the most voted 5 beam sphere on the cloud extracted in the previous step, in order to center the sphere on the nose.
4. Evaluate the mean value of normal vectors of each face vertex with the purpose of obtaining information about face orientation in the three-dimensional space.
5. Move the center of the sphere by the direction evaluated in step 4, and proportionally to the sphere beam value. In this way we collapse the entire sphere in a point that represents the nose-tip.
6. The point that we obtained in step 5 may not be present among the vertex of the examined face. So the last step consists to research the closest face vertex to the nose-tip previously extracted. This vertex represents the real nose-tip.

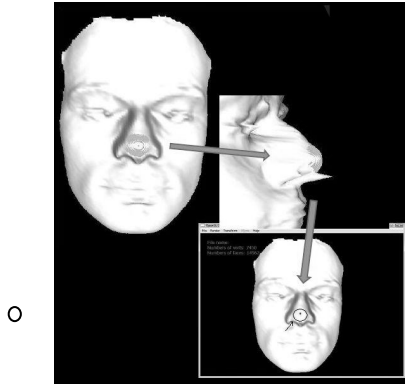


Fig. 7. Example of using 3D-GHT (without pattern scale) to extract nose vertex from a 3D-Face point cloud

Second GHT algorithm was tested with the same 3D-Faces database using only a 3.5 beam sphere as pattern. In this case, the application was able to find spheres with beam of 0.875, 1.75, 2.625, 3.5, 4.375, 5.25, 6.125 and 7. Maximum analysis demonstrated how scale factor introduces in our results both accuracy and new spheres, so the ones next to the nose were not the most voted anymore. We had to consider a different nose-tip research algorithm (it works only if the point cloud represents a frontal face or lightly rotated one) :

1. Calculate mean value of face vertexes;
2. Extract face vertexes that are in a hypothetic sphere of beam 13 (was the value that gave us best results). This sphere is centered in XYZ point calculated in the previous step.
3. Using the latest GHT application with vertexes at step 2 with 1 as threshold value.
4. Evaluate the mean value of normal vectors of each face vertex with the purpose of obtaining information about face orientation in the three-dimensional space.
5. Move the center of the sphere by the direction evaluated in step 4, and proportionally to the sphere beam value in order to collapse the entire sphere in a point that represents the nose-tip.
6. The point that we obtained in step 5 may not be present among the vertex of the examined face. So the last step consists to research the closest face vertex to the nose-tip previously extracted. This vertex represents the real nose-tip.

4.1 Results

18 3D-Faces have been verified. In table 1 we report nose-tip XYZ coordinates calculated by means both algorithms.

As shown in table 1, only in three cases both the algorithms find the same vertex. In other ones, except for faces 2, 3,5 and 14, for each face vertexes found by the algorithms are next each other and both represent a good approximation of nose-tip repere point. Regarding face 5 the second algorithm gives a better result and in face 7

Table 1. Main Results. Note: Coordinates marked with (B) are better in accuracy than coordinates calculated with our other algorithm.

Face	Points' Cloud File*	GavaDB ASE File (Face Source)	Nose-tip XYZ Vertex Coordinates	
			Research through GHT w/o Scale	Research through GHT w/ Scale
1	V1-1.txt	Face01-1.ASE	0.006 -22.914 -556.300	0.006 -22.109 -556.696
2	V1-2.txt	Face01-2.ASE	0.841 -0.829 -566.700(B)	4.186 -2.494 -567.121
3	V1-4.txt	Face01-4.ASE	-10.629 -13.029 -553.1(B)	-10.644 -11.419 -554.106
4	V2.txt	Face02-9.ASE	<i>(0.006 -7.724 -545.446)**</i>	
5	V3.txt	Face03-4.ASE	-92.434 -3.733 -455.800	-95.169 -3.728 -455.350(B)
6	V4.txt	Face04-4.ASE	-11.747 -6.092 -516.200	-12.402 -5.442 -515.991
7	V5.txt	Face05-4.ASE	-----	-95.016 -18.058 -550.200
8	V6.txt	Face06-4.ASE	-92.329 2.590 -548.000	-92.334 1.940 -547.645
9	V7.txt	Face07-4.ASE	-100.244 -19.925 -541.000	-101.039 -19.120 -540.675
10	V8.txt	Face08-4.ASE	-93.292 -8.210 -558.300	-92.462 -9.035 -558.585
11	V9.txt	Face09-4.ASE	-126.019 -20.540 -461.60	-125.189 -18.885 -462.325
12	V10.txt	Face10-4.ASE	-124.921 -14.981 -534.10	-125.716 -16.545 -533.770
13	V11-1.txt	Face11-1.ASE	-147.417 -26.840 -528.30	-147.417 -27.630 -528.495
14	V11-8.txt	Face11-8.ASE	-110.734 -4.060 -552.0(B)	-109.919 -2.435 -551.945
15	V12.txt	Face12-5.ASE	-128.377 0.005 -518.200	-128.372 0.765 -517.925
16	V13.txt	Face13-4.ASE	<i>(-93.656 1.597 -534.485)**</i>	
17	V14.txt	Face14-8.ASE	<i>(-80.260 -26.165 -536.010)**</i>	
18	V15.txt	Face22-5.ASE	-8.454 -6.889 -518.800	-8.449 -6.119 -518.636

**(Both the algorithm use pre-processed points' clouds but in different steps. So we refer to original points' clouds extracted from ASE file associated).*

***(Both the algorithms have extracted the same vertex).*

is able to find nose-tip vertex against failure by the one without scale factor. Considering that the latest research algorithm uses just one GHT run (while the previous version uses two GHT runs), saving about 10 seconds of execution time, we can say we had a good improvement if RAM needed does not represent a problem.

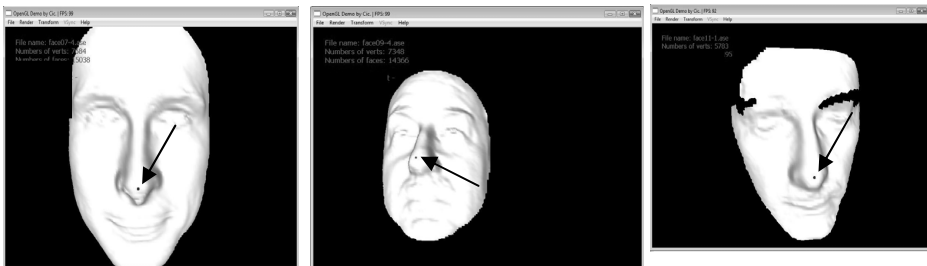


Fig. 8. 3D views of some results: Face 7, Face 9 and Face 11. The subjects are shown in original pose.

5 Conclusions and Future Works

This paper described how we extended Hough Transform for three-dimensional point clouds regarding 3D-face models, and how this GHT can be used in nose-tips detection with good results.

Future work will be lead on:

1. The 3D-GHT, with the purpose of obtaining a low-cost algorithm including pattern rotation.
2. Biometrics applications, combining the 3D-GHT with face geometry in order to improve our nose-tip detection method.
3. Using 3D-GHT to detect other repere point, so as to create a complete AFR System.

References

1. Ballard, D.H.: Generalizing the Hough Transform to Find Arbitrary Shapes. CVGIP 13, 111–122 (1981)
2. Khoshelham, K.: Extending Generalized Hough Transform to Detect 3D Objects in Laser Range Data. Optical and Laser Remote Sensing Research Group, Delft University of Technology, Kluyverweg 1, 2629 HS Delft, The Netherlands; ISPRS Workshop on Laser Scanning 2007 and SilviLaser 2007, Espoo, September 12-14, Finland (2007)
3. Peternell, M., Pottmann, H., Steiner, T., Transform, H.: Laguerre: Geometry for the Recognition and Reconstruction of Special3D Shapes. Institute of Geometry, Vienna University of Technology, Vienna, Austria September 5 (2003)

Face Detection by Means of Skin Detection

Vitoantonio Bevilacqua^{1,2}, Giuseppe Filograno¹, and Giuseppe Mastronardi^{1,2}

¹ Department of Electrical and Electronics, Polytechnic of Bari,
Via Orabona, 4 - 70125 Bari - Italy
bevilacqua@poliba.it

² e.B.I.S. s.r.l. (electronic Business in Security), Spin-Off of Polytechnic of Bari,
Str. Prov. per Casamassima Km. 3 – 70010 Valenzano (BA) - Italy

Abstract. In this article we present a novel approach to detect face in color images. Many researchers concentrated on this problem and the literature about this subject is extremely wide. We thought to decompose the overall problem into two intuitive sub-problems: the research of pixels have skin color in the original image and the analysis of pixels portions resulting, by means of a neural classifier. The achieved results show the robustness of presented approach.

1 Introduction

Face detection is the process has the goal to determine whether or not there are any faces in a digital image and, if present return the image location of each face. This activity is very simple for a man but is very hard for a machine because face detection isn't problem translatable in a mathematic language.

Face detection is a useful task because is a process propaedeutic to facial feature recognition, face recognition e facial expression recognition. All this activities, as a whole, aims to realize video-surveillance system, man-machine interaction and useful robotic application.

The most innovative application about face detection is the realization of integrated circuits that catch faces in a scene in order to get a photo in focus in a photo camera automatically.

In these last years many researchers are developed more than 150 approaches to solve face detection [1]. We can classify this problem in 4 principal methods:

- knowledge-based methods, in which the human knowledge of face is codified in any rules comprehensible to machine;
- template matching methods, that determine image region that have a face with template that describes them entirely or in a part;
- appearance-based method, in which there is an intelligent element that, with a series of examples, recognize a face in different condition;
- feature invariant approaches, that solve the problem finding a structural characteristic of a face (e.g. eye, skin, nose, etc.), in order to find an human face.

Knowledge-based methods are based on intuitive rules that derive by the human observation of a man. Face characteristics are related between them in fact all people have two eyes, a nose, a mouth placed to an opportune distance or that exist a symmetry property on a face. Yang e Wang used a hierarchic knowledge-based method [3].

Their system consists of three rules levels. In the high level, all possible candidate faces are extracted with a window that scans the image. These candidates represent image portions that satisfy the rules of high level, and they are general description of a face by people. In low levels the rules represent the composition of structural components of a face, for example in the central part of face the pixels have a constant value (in gray scale), and there's a relation between central and contour pixels of a face. Beyond these rules the application performs some manipulations on the image. In fact at first level there's a resolution reduction of the input image, then the algorithm executed an histogram equalization of gray levels for potential faces and finally there is an edge extraction. The test on 60 images demonstrates that 50 of this are processed correctly.

Template matching methods find a face applying one or lots of templates that scan the input image. A first attempt of face detection whit this method reported by Sakai et al.[5]. They detect faces by means of two phases: the first aim to localize the regions of interest and the second confirms that these regions are faces with different sub-templates for eyes, nose, mouth and so on. The regions of interest searched transforming the original image in linear segments and performing the match with a template representing face contours. Finally, using each sub-template in each region, they calculated a correlation function to verify the face existence. Analyzing the algorithms realized with template matching, we can note that this technique is unlike for face detection because there is a strong dependence by form, scale and face pose. To delete this problem in these years invented multiscale, deformable and multiresolution sub-templates. Appearance-based methods are based on statistic techniques or on machines learning to find relevant characteristics of faces and non-faces. The best classifier proposed by Viola and Jones, built using AdaBoost learning algorithm [2]. Finally, feature invariant approaches are based on the idea that a man detect face and object in different poses and illumination conditions in a simple mode, so must exists any invariant characteristics on vary revelation conditions. In this method the application determine the facial characteristics first and the face then. Example of facial characteristic are eyes, eyebrows, nose, lips, but also texture (of face, hair and eyes) and skin color. Sirohey have proposed a face localization method to segment a face when there is an heterogeneous background [4]. In this work is used a Canny filter to edge extraction in the input image and a metric to delete any edge and group others. This processing is useful to have a face with only one edge. Finally the application fits an ellipse between head and background. With this method we can notice an accuracy of 80 % on test images. In the Section 2 we present our algorithm. We thought to decompose the overall problem of face detection into two intuitive sub-problems: the research of pixels have skin color in the original image and the analysis of pixel portions resulting, by means of a neural classifier. The solution of the first sub-problem is explained from the sub-section 2.1 to 2.4, while the second sub-problem is presented in the sub-section 2.6. The sub-section 2.5 is the bridge about the two sub-problems because aim to subdivide the just processed image into congruent portions to process called connected components. Finally, the Section 3 outlines the force and weak points of the proposed method.

2 Application Description

The application realized and described in this document, don't belong to anyone of four category presented but it's an hybrid method. In fact, it's an application in part feature invariant and in other part appearance based. It's feature invariant because the first portion of program aim to extract skin patches in a color image, but it's also appearance based because exists a neural face recognizer, that verify if a region hold a face. This application is developed in C++ language and it works whit 24 bit color digital images in any image format. All the functions on image processing manually realized, without any library, but the opening of the images performed about Qt libraries, useful to implement a GUI, too (for more information please visit the URL: [http:// trolltech.com/products/qt/](http://trolltech.com/products/qt/)). The application can be schematized with this block diagram:

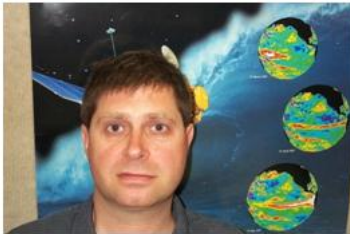
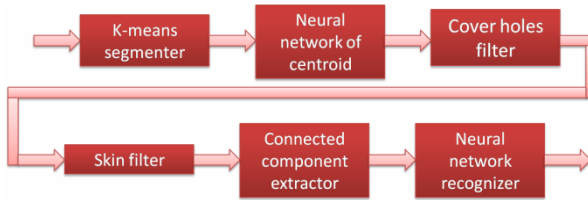


Fig. 1. Input image to K-means segmenter

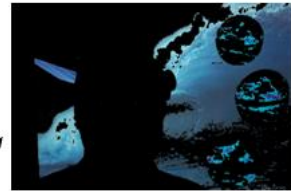


Fig. 2.



Fig. 3.

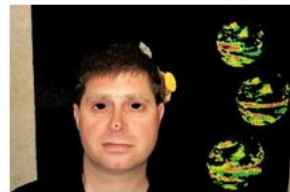


Fig. 4.

Starting by input image at this is made skin detection. Skin detection is the process of extraction of all pixels having skin color. We realized this activity segmenting the image first, and applying a model to describe skin color after. The application then extracts any connected components. For each component are derived some subcomponents, which will process to produce useful output for neural face recognizer. This last module will verify if a subcomponent is a face.

In next subparagraphs we will present theoretically and practically all modules of realized application.

2.1 K-Means Segmenter

K-means segmenter take care of image segmentation, applying a clustering algorithm called k-means. This algorithm starts by hypothesis that the number of cluster, in which will be divided the data set is k . In this case, the data set is the chrominance value of pixels image in CIE-L*a*b* color space. This color space is more useful than RGB because we noticed better output. In fact we performed a series of comparison tests between the two color spaces, that demonstrate how CIE-L*a*b* consents to extract optimally pixel having skin color. The application so operates a conversion from RGB to CIE-L*a*b* color space. In this last space each pixel is represented as combination of three components: L* the luminance component, a* and b* the chrominance components. In k-means segmenter L* component isn't used because it have only luminance informations, while we are interested to chrominance informations. This is another point in favour to CIE-L*a*b* color space because rather than execute k-means algorithm with three chrominance centroids components we used only two components, and this consents to saving useful time. If the segmenter have in input Figure 1, the output consists of three segmented images (Figure 2, 3, 4), of which only one will continue the next computations. Number three is just the value of k . This allow to segmenting in a good manner images that have an uniform background. For heterogeneous background (that are more frequently background) need make other processing, that we'll show in the next sections. K-means algorithm makes clustering based on distance concept of each element by cluster centroid to which it belongs. In various iteration of this segmenter three centroids move up to find a position in which the average of distance of each cluster element by its centroid is minimum. When the movement becomes null the algorithm is stopped and are created the three images (e.g. Figure 2, 3 and 4).

2.2 Neural Network of Centroids

The value of three centroids is the input of a multi-layer perceptron (MLP) neural network feed forward, whose objective is to detect what the three images have more probability to hold a face. This neural network is made up of an input layer of 6 neurons, an hidden layer of 9 neurons and an output layer of 3 neurons. Neural network of centroids have in input the value of three centroids of image and it produces in output the probability that image holds a face. This neural network trained using error back propagation algorithm with in input the centroids of 65 images containing faces.

The validation set formed of centroids of 28 images and the ANN reported always correct results. The values centroids for Figure 1 are:

121.370009	107.760515
124.026731	127.468500
135.179397	158.241178

And the neural network output is:

0.000020	0.024462	0.983285
----------	----------	----------

The image that continues the computation is Figure 4 because it associated to highest neural network output.

2.3 Cover Holes Filter

After the neural network we realized a cover holes filter, that aims to clean entire image deleting the vacancy introduced by k-means segmenter. This filter implemented by means of a series of sliding windows of various dimensions proportional to image area. If contour pixels of window are all coloured and within there are black pixels, then this are re-established to original value. If the input of filter is Figure 4 then the effect is proposed in Figure 5.

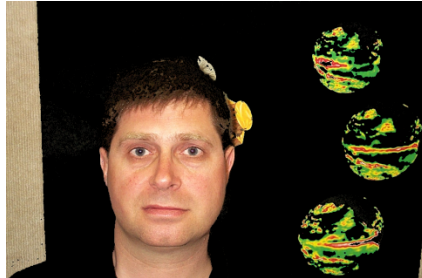


Fig. 5. Cover holes filter output when in input there is Figure 4

2.4 Skin Filter

Skin filter consists of implementation of elliptical boundary model for skin color. This model describes in a good manner the skin color distribution in all chrominance space by means of two parameters [6]. The inventors of this model verified that the skin color distribution in opportune chrominance spaces (r-g normalized and CIE-xy) very well approximated by an ellipse. We realized this model implementing a three-dimensional histogram in r-g normalized space, obtained counting chrominance values of pixels in 44 example skin patches (Figure 6).

In Figure 7 there is the 3D histogram obtained and in Figure 8 there is the same histogram seen in an up view. In this second view we can see that skin color pixel



Fig. 6. Skin patches used for implementation of three-dimensional histogram

thicken in an elliptical zone. If we indicate with X the chrominance value of pixel, the ellipse can be described by this formula:

$$\Psi(X) = (X - \Psi)^T C^{-1} (X - \Psi) \tag{1}$$

Where Ψ and C are the average values of chrominance vectors and covariance matrix respectively. The values of Ψ and C are this:

$$\Psi = \begin{bmatrix} 0.495271 \\ 0.25982 \end{bmatrix} \qquad C = \begin{bmatrix} 0.0036 & -0.0015 \\ -0.0015 & 0.0013 \end{bmatrix}$$

We tested the skin filter with a series of images discovering that the best threshold value that consents to reject pixels whose color is different by skin color is $\theta=5$. If $\Phi(X)$ is less than θ then pixel is skin, or else is blackened. Equality $\Phi(X)=\theta$ identifies an ellipse in r-g normalized color space, when Ψ and C are the center and ellipse principal axes. In Figure 9 is possible to analyze the effect of skin filter when the input is Figure 5. The elliptical boundary model isn't the only study about skin detection, similar works are developed by Terrillon et al.[7] have used single gaussian skin model and by Yang and Ahuja[8] that have implemented the mixture of gaussian skin model. We've preferred elliptical boundary model because at the state of art it's the

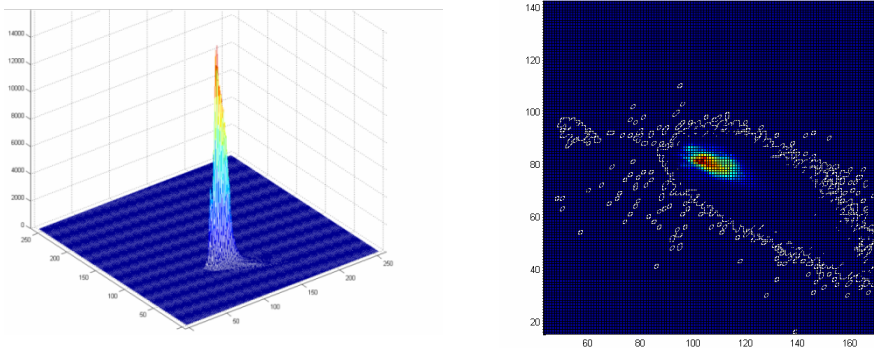


Fig. 7. Three-dimensional histogram of skin distribution and its up view



Fig. 8. Skin filter output

best model for skin detection. However for a complete presentation of skin color model you can refer to Yang, Kriegman et al.[9].

2.5 Connected Component Extractor

After skin extraction, we can search connected component in image, but, to make this, the image must be transformed in binary. So color pixels are set to 1, while black pixels to 0. A connected component is an image portion composed of a pixels set that verify 4-connection property [10]. A pixel is 4-connected if at least one of its 4 near pixels (placed in right, left, up and down) are set to 1. Really all components didn't considerate, because also lone pixel acts as connected component. So we have introduced a check on components area, which must be above the 0.05 % of image area so that the component can proceed in next computations. We've chosen to delete components that have this area value because it consents to delete the noise visible in Figure 8. At this point the application sequentiality stops for introduction of some thread. In fact for each connected component we created a thread that takes care of: extract some sub-components in square for, magnify or reduce them to make them multiple of a window 20x20 and finally equalize gray levels histogram to amplify some shades. Moreover for each sub-component we calculated 404 means with two sliding window with different form, so that it occupies 404 position. These positions are visible in Figure 9 and for the second face we can note that the average is calculated on 4 rows blocks. This is made because the 4 averages contain the fact that the eyes and mouth are more dark than central part of a face.

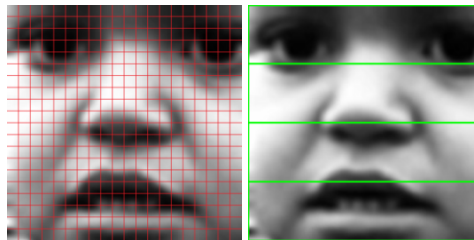


Fig. 9. Positions of sliding window when application calculate 404 means

2.6 Neural Face Recognizer

The neural face recognizer is a MLP neural network able to detect if a subcomponent is a face. Its architecture is made of an input layer of 404 neurons, a hidden layer of 200 neurons and an output layer of a neuron, and all neurons have as activation function an hyperbolic tangent sigmoid function. The network accepts in input 404 means calculated for each subcomponent and it produces in output the probability that this subcomponent is a face.

This network is trained with face and non-face examples (see Figure 10), using error back propagation algorithm. In fact we have used 80 different images face at various dimension for a total of 719 images, while for non-faces we have employed 441 examples contained skin portions of leg, hand, neck and so on. The calculated means are normalized in interval $[-1,1]$ to represent congruent input for neural network. If the output of recognizer overcomes the threshold value 0.8 than the thread will apply in original image a colored square. This threshold value allows to reach trade-off between false positives and false negatives.



Fig. 10. Examples of face and non-faces used to learn Neural Face Recognizer

3 Test and Conclusion

The application is tested on a set of 50 images searched casually on the web. In this set 10 images have only one face, 7 no face and 33 are group photos. The neural network of centroid work good in 96 % of cases, so only 2 images on 50 aren't analyzed correctly by next modules. Excluding this two images the test results are:

- detection rate of 64.04 % and detection rate of 90 % for images containing one face (this means that 9 face on 10 are detected);
- 36 % of false negatives and 35, 96 % of false positive (i.e. 67 false negatives on 48 test image).

Compared to literature we can derive that application detected faces in a good manner in optimal condition. For optimal condition we intend uniform background and/or face at intermediate distance by camera. Doing a comparison with developed application by Rowley et al.[11], that used MLP neural network, he obtain a detection rate of 90 %, having used 1050 face and 8000 non-faces for training. In our application we have 66 % of detection rate using 80 faces and 441 non-faces. Future developments will provide an improvement of detection rate and a reduction of false positive. Initially this application didn't support threads and execution was slow. Execution time is a crucial factor in video surveillance application or in robotics. In our case this time depends on image to process and on number and extensions of connected components. With threads

introductions execution time is lowered, saving from few seconds, for images whose execution time is 25-30 seconds, to 16 seconds, for images more complex whose execution time is 86-120 sec. We have chosen ANN as classifier just to reach a trade-off between detection rate and this important parameter. We must spend few words about skin detection technique used. The fact of skin pixel are detected with first four application blocks allow us to realize the skin filter using a number really low of skin patches, e.g. 44 pictures versus 4675 images of Lee[6]. In fact the presence of k-means segmenter consents to reject most of background pixel in order to allow a refining work for skin filter. Next table summarizes comparisons relating to Rowley algorithm [11] and our method.

Figure	Dimensions	Execution time (s)	Detection Rate (%)		False Positives		False Negatives	
			Rowley	Bevilacqua	Rowley	Bevilacqua	Rowley	Bevilacqua
11	800x600	19	100	100	0	0	0	0
12	1024x768	39	100	100	0	1	0	0
13	1024x890	77	55.56	66.67	0	3	4	3
14	1024x769	89	86.36	68.18	0	6	3	6
15	800x600	292	0	29.03	0	0	31	22



Fig. 11. Comparison between Rowley (left picture) and our algorithm (right picture)



Fig. 12. Comparison between Rowley and our algorithm

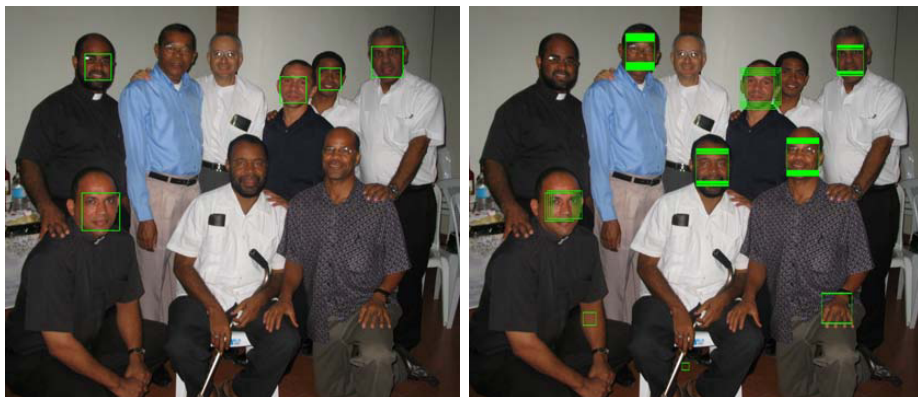


Fig. 13. Comparison between Rowley and our algorithm



Fig. 14. Comparison between Rowley and our algorithm



Fig. 15. Comparison between Rowley and our algorithm for very small faces

References

1. Yang, M., Kriegman, D., Ahuja: Detecting Faces in Images: A Survey. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 24(1) (2002)
2. Viola, P., Jones, M.: Robust Real-Time Face Detection. *International Journal of Computer Vision* (2004)
3. Yang, G., Huang, T.S.: Human Face Detection in Complex Background. *Pattern recognition* 27(1), 53–63 (1994)
4. Sirohey, S.A.: Human Face Segmentation and Identification. Technical Report, University of Maryland (1993)
5. Sakai, T., Nagao, M., Fujibayashi, S.: Line Extraction and Pattern Detection in a Photograph. *Pattern Recognition* 1, 233–248 (1969)
6. Lee, J., Yoo, S.: An Elliptical Boundary Model for Skin Color Detection. In: *International Conference on image science* (2002)
7. Terrillon, J., Akamatsu, S.: Comparative Performance of Different Chrominance Spaces for Color Segmentation and Detection of Human Faces in Complex Scene Images (1999)
8. Yang, M., Lu, W., Ahuja, N.: Gaussian Mixture Model for Human Skin Color and its Application in Image and Video Database. In: *Conference on Storage and Retrieval for Image and Video Database* (1999)
9. Yang, M., Kriegman, D., Ahuja, N.: Detecting Faces in Images: A Survey. *IEEE Transaction on pattern analysis and machine intelligence* 24(1) (2002)
10. Gonzalez, R., Woods, R.: *Digital Image Processing*. Prentice-Hall, Englewood Cliffs
11. Rowley, H., Baluja, S., Kanade, T.: Neural Network-Based Face Detection. *IEEE Transaction on pattern analysis and machine intelligence* (1998)

Research on Product Case Representation and Retrieval Based on Ontology

Yan Shen, XiaoPeng Wei, and JianWei Wang

Liaoning Key Lab of Intelligent information Processing,
Dalian University, 116622 Dalian, China
shenyan0417@163.com

Abstract. Aiming at the problem that no representation can realize completely expression, intelligibility and easy operation, a method of knowledge representation based on ontology is adopted and it is applied to product case. The discrimination of case similarity is the foundation of case retrieval. Considering the influence on similarity from attribute, an improved similarity algorithm is presented. Hierarchical retrieval based on attribute tables is used. Then the best case is properly searched. Finally, a cleaner as an example is provided to illustrate the validity about the case retrieval.

Keywords: ontology; knowledge representation; case retrieval; similarity.

1 Introduction

It is the knowledge economy time that the economy tends to globalization. More intensely the world market compete, more countries pay attention to the development of the new products. Knowledge representation for product conceptual design determines not only application form but also how to deal with the knowledge efficiently for representation methods.

There are a lot of studies on knowledge representation methods. Hsiao-chen You[1] studies the significance of demands between customers and products with the semantic description for product design. But this representation is verified in theory without the specific application. It is limited to theoretical analysis. Feng Hao[2] introduced the Object-oriented representation which the abstraction, the encapsulation and the inheritance is combined to classify the product knowledge. PeiGen Liu [3] proposes an extended production rules which the premise, the conclusion and the process of the rule are expressed with conceptual graphs. However, these methods have some shortages of functional description for product design. It is the main problem that how to express the field knowledge completely, operate and maintain it easily.

Based on the above problem, a method of knowledge representation based on ontology is adopted. Ontology is an explicit, partial account of a conceptualization or the intended model of a logical language [4]. There are four advantages of using knowledge representation based on ontology: 1) A conceptual model, which the related concepts can be extracted from objective world to establish the model. The meaning of concepts is independent of environmental condition. 2) Definition, which the concepts and constraint is clearly defined. 3) Formalization, which it is operated on computer.

4) Sharing, which the common comprehension of the knowledge is realized. Moreover, ontology is a good knowledge carrier. Not only does it provide formal description, but also it realize knowledge sharing. That is useful to organize product knowledge and retrieve cases. Therefore, if ontology is adopted to express and organize product knowledge it will be useful for case retrieve. The matching information based on ontology is more specific, so that it can improve the efficiency of retrieval case.

The remainder of this paper is organized as follows: Section 2 provides a method of product representation based on ontology, which is expressed by 5-tuple: (C,I,R,F,A). We take a cleaner as an example to illustrate the procedure of representation. In Section 3, an improved algorithm of case retrieval base on attribute table is introduced. By similarity calculations, the best case is searched. In Section 4, we illustrate the feasibility and the validity about the case retrieval. Finally, in Section 5 some conclusions are pointed out.

2 Product Case Representation Based on Ontology

Ontology is an art of organizational information. It is relatively stable and independent of applications. So it is used by human and computers to realize the purpose that information can be shared and reused. Before the ontology models are constructed, these entities involving concept, instance, relation, axioms must be specially explained. Most of words can describe these entities which are confirmed. Normally, the frame structure includes concept names, relation sets and concept descriptions with natural language.

Gruber[5] gives the classic definition of ontology, which can be expressed by 5-tuple: (C,I,R,F,A), where C is the set of concepts that is used to describe objects. I is the concept example which is the element and the object in semantic level. R is the set of relation based on the set of concepts. There are four relations between concepts: part_of, kind_of, instance_of and attribute_of, where both part_of and kind_of are the same as the relation between class and subclass in object-oriented; attribute_of is a concept of the attribute of another. F is a function relation also based on the set of concepts; A is the axiom that is used for constraining the concept, relation and function. If the construction of knowledge ontology is easy, it can be expressed by 3-tuple(C, R, A).

Take a cleaner as an example to realize its function and structure representation with 3-tuple(C, R, A). Definitions are as follows:

```

cleaner_ontology(Ccleaner, ACcleaner, Rcleaner)
Ccleaner={ Supply,Transmission,Clean,Leach,Input,Rotation,Electromotor,Exhaust_fan,Clean_box,Flexible_hose,Long_hose,Soft_hose };
ACcleaner={ Power,Weight, Handle,Cubage,Exhaust way };
ACcleaner(Power)={ 1200-1500w };
ACcleaner(Weight)={ 5.0kg };
ACcleaner(Handle)={ double };
ACcleaner(Cubage)={ 3L };
ACcleaner(Exhaust way)={ up };
Rcleaner={ Part_of, Attribute of };

```

where, Ccleaner is the set of concepts that is used for clean descriptions; A^Ccleaner is used for constraining the concepts of cleaner, relation and function; Rcleaner is the relation based on the concepts of cleaner.

This definition based on ontology can be realized by computer. According to the relation and rule defined, the set of concepts can be automatically generated a tree-structure with representation language. The tree-structure of ontology representation about cleaners with OWL is shown in Fig.1.

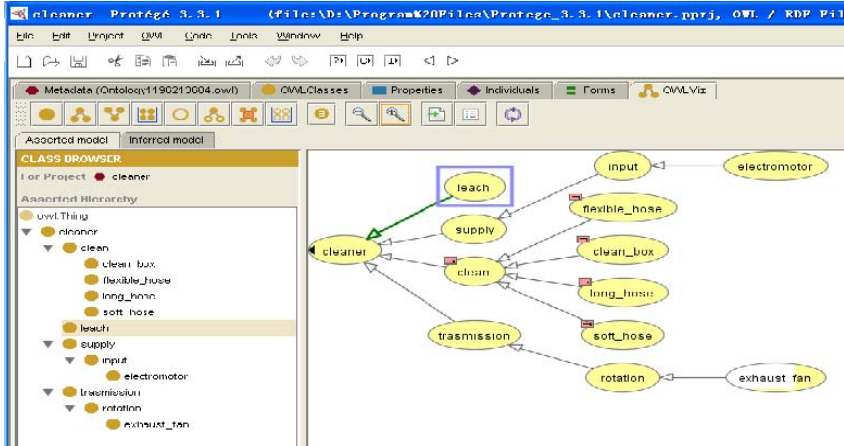


Fig. 1. The structure of ontology representation about the cleaner

3 Hierarchical Retrieval Based on Attribute Table

Similarity calculation is the foundation of case retrieval. It is the key content to calculate similarity between two cases.

3.1 Similarity Calculation

Definition: The similarity is a metric of attributes between two product cases. Given two cases X and Y. Sim(X, Y) is defined as the similarity between X the Y within the range [0,1], where Sim is a function that returns 1 if X and Y are completely the same or 0 otherwise. Normally, the formula of similarity calculation based on attributes is as follows:

$$Sim(X, Y) = \frac{N_{Requirements}^{Cases} \times 2}{N_{Requirements} + N_{Cases}} \tag{1}$$

where, $N_{Requirements}^{Cases}$ is the matching number between attributes of cases and requirements. $N_{Requirements}$ is the number of the attribute of requirements. N_{Cases} is the number of the attribute of cases.

3.2 An Improved Similarity Calculation

A good similarity algorithm will improve the accuracy of results. For this purpose, we introduce here three similarity calculation formulas under different conditions. The similarity calculation formulas based on attributes are as follows:

$$Sim(A, B) = 1 - \frac{|B - A|}{\beta - \alpha} \tag{2}$$

where, the value of A and B is within the range $[\alpha, \beta]$.

$$Sim(A, [B_1, B_2]) = \frac{\int_{B_1}^{B_2} Sim(A, x) dx}{B_2 - B_1} = \frac{\int_{B_1}^{B_2} (1 - \frac{|x - A|}{\beta - \alpha}) dx}{B_2 - B_1} \tag{3}$$

where, the value of A, B_1 and B_2 are within the range $[\alpha, \beta]$ and the value of x is within the range $[B_1, B_2]$.

$$Sim([A_1, A_2], [B_1, B_2]) = \frac{\int_{A_1}^{A_2} \int_{B_1}^{B_2} Sim(x, y) dy dx}{(A_2 - A_1)(B_2 - B_1)} \tag{4}$$

where, the value of x is within the range $[A_1, A_2]$ and the value of y is within the range $[B_1, B_2]$.

The total similarity of the case is

$$Sim = \sum_{i=1}^N Sim_i W_i \tag{5}$$

where, Sim_i is the similarity of the i th attribute; W_i is the weight of the i th attribute.

3.3 The Algorithm of Hierarchical Retrieval Based on Attribute Table

According to the weight of the matching attributes, it is classified to N hierarchies to matching calculate. The purpose is that the range of search is shrunk and the efficiency of retrieval is increased. Finally, calculating the total similarity of cases is used to compare the values. The best case is determined by the maximum of the total similarity. The process of searching is finished.

There are three features about the algorithm of hierarchical retrieval: (1) Attribute tables, which is a table of descriptions for the mainly characteristic attribute from the cases. The characteristic attribute is matched to calculate similarity. (2) Weighted method, which each characteristic attribute is assigned to weight. The larger is the value of weight, the more important is the characteristic attribute. The value of weight is within the range $[0, 1]$. (3) Hierarchical retrieval, which is retrieved by N hierarchies. The characteristic attributes are ordered by the value of weights. Calculate the similarity of the characteristic attributes and the total similarity by N hierarchies. It is realized that the range of matching is shrunk. Moreover, the retrieval efficiency is improved.

The descriptions of algorithm are as follows:

- Step1: The design requirements set $R=\{f_1, f_2, \dots, f_n\}$ is input. The corresponding weight is $W_i, i=1, 2, \dots, n$.
- Step2: The requirements are ordered by $W_i: f_1 > f_2 > \dots > f_i \dots > f_n$. In the case attribute table, the existing design requirement f_i is selected to match.
 - (1)When $i=1$, each case is searched in the attribute table. The characteristic attributes of the case vector are $\{E_1, E_2, \dots, E_m \dots E_n\}$;
 - (2)When $i > 1$, each case is searched in C_{i-1} . Here, C_i is the subset of cases.
- Step3: Calculate similarity.
 - (1)The characteristic attributes of the case E_m , which is corresponding with the requirements, is selected from attribute table or C_{i-1} .
 - (2)Compare with f_i with E_m . Calculate the characteristic attribute similarity Sim_i and the total similarity Sim_{k_i} .
 - (3)Repeat (1) and (2) until that each case similarity is calculated.
- Step4: Judge the value of i .
 - (1)If $i < n$, go to Step 5;
 - (2)If $i = n$, go to Step 8.
- Step5: Compare all the similarity: $Sim_{k_1}, Sim_{k_2}, \dots, Sim_{k_n}$. Order them.
- Step6: T_s , which is a threshold value, is assigned to 0.5. If $Sim_{k_i} \geq T_s$, its case is selected to store in the set of C_i .
- Step7: Let $i=i+1$, go to Step 2.
- Step8: The best case will be determined by the maximum of total similarity.

4 Experiment

In the Section 2, a case based on ontology is provided. It is used as a design requirement to matching. Its attributes of the design requirements are shown in Table 1. The characteristic attributes of a series of cleaners are shown in Table 2. The attributes in Table 1 will be matched with the attributes in Table 2.

The retrieval results according to the requirements are shown in Table 3. The shadow is the similarity which is more than or equal to 0.5.

Table 1. The attributes of the design requirements

No.	The characteristic attribute	W_i	The value of attribute
1	Weight	0.35	5.0kg
2	Cubage	0.25	3.0L
3	Power	0.2	1200w-1500w
4	Handle	0.15	double
5	Exhaust way	0.05	up

Table 2. The characteristic attributes of a series of cleaners

No.	1	2	3	4	5	6	7	8
ID_ Instance	CG461	CG463	CG661	CG663	CL673	CG677	E7111	E7113
Exhaust way	up	up	up	up	up	up	normal	normal
Handle	double	double	double	double	single	double	single	single
Weight (kg)	4.5	4.5	4.7	5.1	5.1	6.7	4.1	4.1
Power cord (m)	5.0	5.0	6.0	4.5	5.0	6.0	5.0	5.0
Power (w)	1300	1500	1500	1600	1800	1200	1200	1300
Rev (r/s)	500	500	500	500	500	500	500	500
Suction (kpa)	18	18	20	25	25	30	15	15
Cubage (L)	3.0	3.0	3.2	2.5	3.0	2.5	1.8	1.8

Table 3. The retrieval results according to the requirements

Input								
No.	1	2	3	4	5	6	7	8
Weight	4.5	4.5	4.7	5.1	5.1	6.7	4.1	4.1
Cubage	3.0	3.0	3.2	2.5	3.0	2.5	1.8	1.8
Power	1300	1500	1500	1600	1800	1200	1200	1300
Handle	double	double	double	double	single	double	single	single
Exhaust way	up	up	up	up	up	up	---	---
Output Similarity(Sim _i)								
Weight	0.8077	0.8077	0.8846	0.9615	0.9615	0.3462	0.6538	0.6538
Cubage	1.0000	1.0000	0.8571	0.6429	1.0000	---	0.1429	0.1429
Power	0.7222	0.5000	0.5000	0.1667	0.4863	---	---	---
Handle	1.0000	1.0000	1.0000	---	---	---	---	---
Exhaust way	1.0000	1.0000	1.0000	---	---	---	---	---
Sim _{ki}	0.8771	0.8327	0.8239	---	---	---	---	---

The specific process of retrieval is as follows:

(1) The design requirements are ordered according to the value of weight. The first one is Weight. Calculate its similarity with Eq. (2). The similarity is more than or equal to 0.5 is C1 (1,2,3,4,5,7,8). No.6 will be not calculated any more.

(2) The second one is Cubage. Calculate its similarity with Eq. (2). The result is C2(1,2,3,4,5).

(3) The third one is Power. Calculate its similarity with Eq. (3). The result is C3(1,2,3).

(4) The attribute of Handle is matched. The result is C4 (1,2,3,).

(5) The attribute of Exhaust way is matched. The result is C5 (1,2,3,).

Finally, according to Eq. (5), compare the total similarity calculation. The largest one is Sim1. Therefore, the best case is CG461.

The curve of improved similarity algorithm is shown in Fig.2. Date 1 is the improved similarity. Date 1 is more accurate that of Date 3.

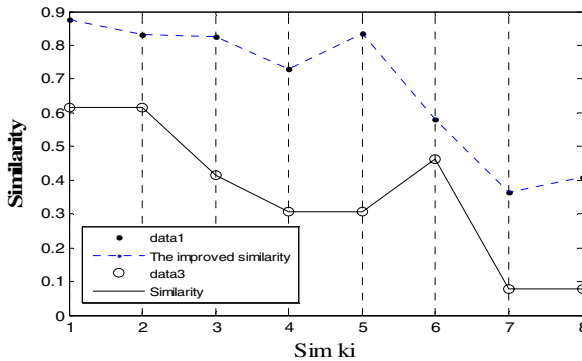


Fig. 2. The curve about compared the two similarity algorithms

The best case is determined by comparing the total similarity. Sim_{k_i} is the total similarity of each case. With the improved similarity calculation, the results of experiment illustrate that case retrieval is more accuracy than before. The case which is most similar with the design requirements is searched.

5 Conclusion

(1) A method of product presentation based on ontology is proposed. The method can realize not only standardization of expressive information but also normalization of functional knowledge for conceptual design.

(2) In the process of case retrieval, hierarchical retrieval based on attribute tables is adopted. By the similarity calculation based on attributes and the total similarity calculation, the range of matching is shrunk. Moreover, the retrieval efficiency is improved. The result is more accurate so that the best case can be searched.

(3)The experiment results show that hierarchical retrieval based on attribute tables is practicable and available.

Acknowledgements. The project is supported by the National Natural Science Foundation of China (grant No. 50575026), Excellent Youth Talents Foundation of Liaoning Province (No. 3040014) and open funds of Liaoning Key Lab of Intelligent Information Processing, Dalian University.

References

1. You, H.C., Chen, K.: Applications of Affordance and Semantics in Product Design. *DesignStudies* 28(1), 23–38 (2007)
2. Feng, H., He, Y.L., Ma, F.L., et al.: Application of Object-oriented Representation in the Motorcycle Designed Expert System. *Journal of Chongqing University* 28(1), 1–4 (2005)
3. Liu, P.Q., Li, Z.Z., Zhao, Y.L.: Knowledge Representation of Extended Production Rule. *Journal of Xi'An Jiaotong University* 38(6), 587–591 (2004)
4. Studer, R., Benjamins, V.R., Fensel, D.: Knowledge Engineering, Principles and Methods. *Data and Knowledge Engineering* 25(1), 161–197 (1998)
5. Gruber, C.T.R.: A Translation Approach to Portable Ontologies. *Knowledge Acquisition* 5(2), 99–220 (1993)

Retinal Fundus Biometric Analysis for Personal Identifications

Vitoantonio Bevilacqua^{1,2}, Lucia Cariello^{1,2}, Donatello Columbo¹,
Domenico Daleno^{1,2}, Massimiliano Dellisanti Fabiano³, Marco Giannini⁴,
Giuseppe Mastronardi^{1,2}, and Marcello Castellano^{1,2}

¹ Department of Electrical and Electronics, Polytechnic of Bari,
Via Orabona, 4-70125 Bari- Italy

{Bevilacqua, Mastrona, Castellano}@poliba.it,
{Cariello, Daleno}@deemail.poliba.it

² e.B.I.S. s.r.l. (electronic Business in Security), Spin-Off of Polytechnic of Bari,
Str. Prov. per Casamassima Km. 3-70010 Valenzano (BA) -- Italy

³ Digital Future Engineering,
Via G. Petroni, 117/B – 70124 Bari – Italy

mdf@dfuture.it

⁴ Centro Laser s.c.a.r.l.,
Str. Prov. per Casamassima Km. 3 – 70010 Valenzano (BA) – Italy

marco.giannini@centrolaser.it

Abstract. In this paper a biometric system for personal identification, realized through the manipulation of retinal fundus images and the detection of its bifurcation points, is described. In the image pre-processing step, a strong contrast exaltation between blood vessels and the background in retinal image is carried out; then blood vessels are extracted and next the vasculature bifurcation and crossover points are identified within squared shaped regions used to window the image. Finally the features sets are compared with a pattern recognition algorithm and a novel formulation is introduced to evaluate a similarity score and to obtain the personal identification.

Keywords: personal identification; retinal fundus; blood vessels detection; vasculature bifurcation and crossover points extraction; clustering algorithm; Naka-Rushton filter; Generalized Hough Transform; pattern recognition.

1 Introduction

The retina is located inside the eyeball on its back wall (Fig.1). Blood reaches the retina through vessels coming from the optic nerve. Just behind the retina is a matting of vessels called the choroidal vasculature. The mat of vessels of the choroid just behind the retina reflect most of the useful information used to identify individuals. This area of the eye is also referred to, by medical doctors, as the retinal fundus [1].

The retinal pattern capturing process can be summarized as follows this: the user looks at a green dot for a few seconds until the eye is sufficiently focused for a scanner to capture the blood vessel pattern. An area known as the fovea, located in the

centre of the retina, is then scanned by an infrared beam [2]. Due to its internal location, retina is protected from variations caused by exposure to the external environment (as in the case of other physical characteristics used in personal identification like fingerprints, palm-prints etc.).

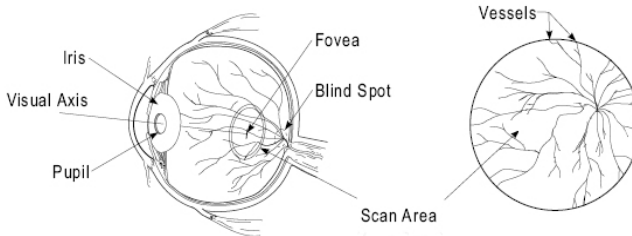


Fig. 1. Eye and scan area

Awareness of the uniqueness of the retinal vascular pattern dates back to 1935 when two ophthalmologists, Drs. Carleton and Goldstein, while studying eye disease, made a startling discovery: every eye has its own totally unique pattern of blood vessels. They subsequently published a paper on the use of retinal photographs for people identification based on blood vessel patterns [3]. Later in the 1950's, their conclusions were supported by Dr. Tower in the course of his study of identical twins [4]. He noted that, considering a couple of persons, identical twins would be the most likely to have similar retinal vascular patterns. However, Tower's study showed that, among resemblance factors typical of twins, retinal vascular patterns are the one that have the least similarity. Besides, among human physical features, none is longer lasting than retinal fundus. Consequently, a retinal image offers an extremely accurate personal identification tool [5].

This paper describes a methodology to perform personal identification by detecting bifurcation points of retinal fundus vasculature. The process consists of three phases: at first each retinal image is handled by pre-processing operations aiming to reduce its intrinsic noise; then, blood vessels are extracted to allow, subsequently, the detection of the bifurcation points necessary to personal identification.

A database of 12 images containing 256 grey levels obtained by the ophthalmic clinic of a local hospital, has been used for testing the algorithm. The digital medical images constituting this database, were acquired by fluorescent angiography and, moreover, have a resolution of 512 x 512 pixels and a quantization of 8 bit per pixel.

2 Pre-processing

Through the pre-processing stage, two goals are to be reached: in primis a reduction of image noise, coming both from the disuniformity of light radiation on retinal fundus and from image acquisition hardware itself, and in secundis, a contrast enhancement of the lines that describe blood vessels, to simplify the successive vasculature extraction.

2.1 Noise Reduction

The first target has been successfully achieved by clustering the hole image in regions depending on the mean grey value of each area with a binarization operation (Fig. 2). Binarization is performed using two different selection criteria, the first one based upon a numeric approach, that considers the percentage of bright pixel of the image, and a second one based upon chromatic informations that considers, also, the gray value of the pixels previously calculated. The selection criterion based on the percentage of bright pixel allows to improve the determination of bright areas in case of overexposed images during the data capture step. The lower threshold, that shows the lowest gray value represented in a bright region, used in the second criterion, has been located considering the shift that afflicts the histogram of images acquired under different illumination conditions. Finally, the 40% of bright pixels of the images, belonging to the range [160, 255], will be selected. At this point, the necessity of applying the Naka-Rushton filter to every single region reveals clear. The use of this filter, in fact, allows to obtain a strong contrast enhancement between blood vessels and background, since each pixel belonging to areas under different illumination condition will be processed after being weighted by the average of pixels having the same brightness level, decreasing considerably the noise.

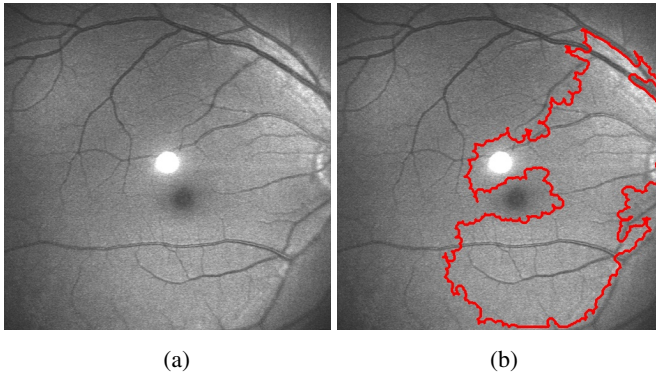


Fig. 2. Retinal Fundus (a), Bright Region (b)

2.2 Contrast Enhancement

The second goal, as introduced before, has been accomplished by applying Naka-Rushton filter to both the regions defined in the previously step. The Naka-Rushton technique, that consists of a strong compactness operation of grey levels [6] [7], is regulated by the law:

$$O(i, j) = \frac{I(i, j)}{I(i, j) + \mu_{window}} \quad (1)$$

where $O(i, j)$ is the output matrix, that is the transformation result, $I(i, j)$ is the elaborated image matrix and μ_{window} is the mean gray value of the pixels in the exploration window.

The result of this process is an image with a greater contrast between background and objects, as depicted in Figure 3.

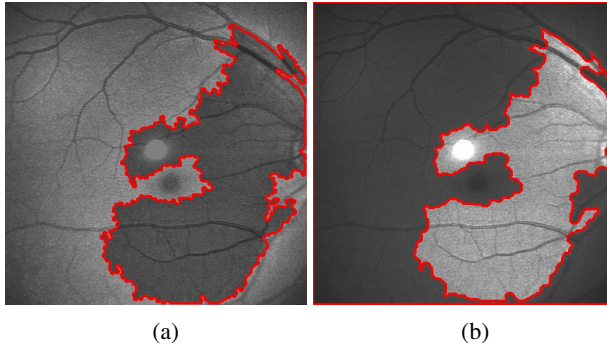


Fig. 3. Bright region after Naka - Rushton filtering (a), dark region after Naka - Rushton filtering (b)

3 Blood Vessel Detection

A blood vessel appears, as a line that is darker than the background, within the image.

The lines extraction algorithm, developed to find out blood vessels, is based on the image filtering, by means of a gaussian mask that allows the enhancement of the points of discontinuity within the image. The reason behind the choice of a kernel of Gauss resides in the necessity of performing a smooth operation on the image in order to reduce noise [8].

The image has been derived by convolving it with gaussian kernels and with the first and second derivative of the same kernel, and the position of the line crests has been determined thanks to a sub pixeling algorithm. This algorithm discovers the direction that is orthogonal to the line by calculating the first derivative of image, and the exact position of the center of the line by determining the maximum value for the second derivative in that area.

Analytically, each line can be described as follows:

$$r(x) = \begin{cases} a, & x < -\omega \\ h, & |x| \leq \omega \\ b, & x < \omega \end{cases} \quad (2)$$

where:

- $(a,b) \in [0,h]$ are coefficients representing the chromatic differences on the edge of a generic line;
- $h \in [0,255]$ is the maximum grey value that the line assumes in the image;
- 2ω represents the line thickness.

The existence of the line is determined by studying the first spatial derivative of the image. The pixels where the first derivative has a null value are considered as belonging to the border of a line. But this is not enough: the noise that affects the pictures of retinal fundus creates false positives making, for this reason, impossible the extraction of lines by means of a bare calculation of the first derivative. The adopted criterion to overcome this disadvantage, is based upon the study of the second derivative behaviour in the region laying around the points where the first derivative is zero. The exact position of the line center is determined searching for the second derivative maximum value in that area, with a sub pixel precision.

The operation of image derivation is computed as the result of the convolution of the image itself with Gauss kernel built as in Figure 4.

-1	-1	0	1	1	-1	-1	0	-1	-1	0	1	1	1	1
-1	-1	0	1	1	-1	-1	0	-1	-1	-1	0	1	1	1
-1	-1	0	1	1	0	0	0	0	0	-1	-1	0	1	1
-1	-1	0	1	1	1	1	0	1	1	-1	-1	-1	0	1
-1	-1	0	1	1	1	1	0	1	1	-1	-1	-1	-1	0

Fig. 4. The used Gauss Kernel

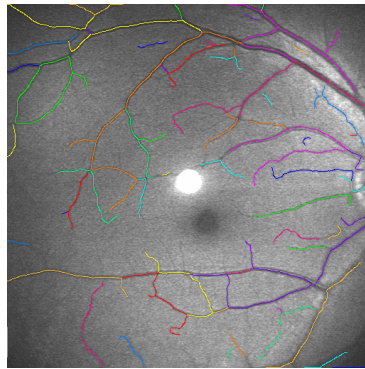


Fig. 5. Detected blood vessels

The gaussian filter will be more efficient while increasing the dimensions of the convolution matrix. The dimension of the used filter has been determined considering the dimension of the thinnest blood vessel to be detected (Fig. 5).

4 Bifurcation Points Extraction

The research of blood vessels crossing points has been developed studying of intersections of lines extracted in the previous phase, within squared shaped regions used to window the image.

In fact the image has been windowed by using squares with 8 pixel side, that moves along the principal axis of the image so that two consecutives squares overlap for the half of their area (Fig. 6).

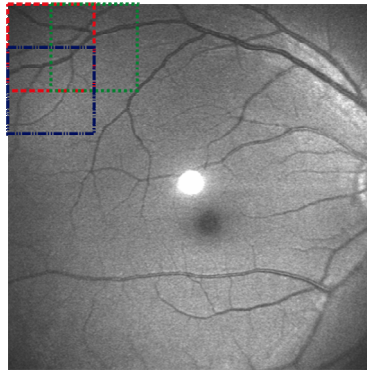


Fig. 6. Squared shaped regions used to window the image

The study of map of extracted lines has been limited to the area enclosed inside the window. Segments within each window are considered as straight lines and their intersection is geometrically computed. The risk of losing intersection points eventually located on the boundary of the region is avoided by choosing windows that partially overlap. Within each region only the points which falls inside the window boundaries are considered. All the intersection points falling outside the window will be computed and stored when the algorithm will investigate the region to which they belong.

The windows positioning criterion brings to overlap of solutions: the same intersection point can be determined more than once, i.e. one time for each window positioned in that area.

The result is that we have more points that represent the same intersection of vessels, positioned around the real intersection point (Fig. 7 (a)). To calculate the exact coordinates of the intersection points, the average value for both the coordinates x and y , has been calculated (Fig. 7 (b)). Another error is introduced where interpolating curves with straight lines. The interpolation error has been reduced by considering smaller windowing squares.

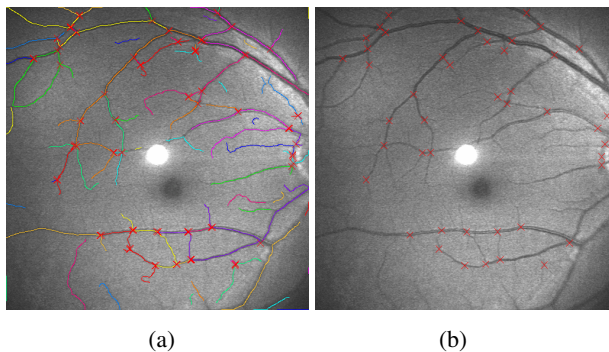


Fig. 7. The extracted cloud of points (a); the final extracted blood vessels crossing points (b)

5 Personal Identification

The cloud of points, extracted as described, represents a feature set which identifies each retina (i.e. each subject). In this step the goal is to use a matching algorithm which provides a score of correlation between two clouds of points (i.e. our features sets). Each feature is represented by a couple of coordinates, hence a simplified version of the Generalized Hough Transform [9] has been tested. The Generalized Hough Transform is an effective way to recognize shapes: it makes a comparison between two set of connected points by means of an accumulation matrix (AM). Each couple of connected points of one set is compared with each couple which shows the same inclination on the other set. Since the used set does not consist of connected points, we use a simplified algorithm which can be described as follows:

```

Calculate RA and RB (reference points for features set A and B. It
can be arbitrary set)

Refer all points of each features set to their own reference point

For each point Pa of set A
  For each point Pb of set B
    Calculate distance between Pa and Pb and add contribute in
    the accumulation matrix

```

A peak in the AM will represent a high factor of similarity (Fig. 8 (a)). Of course noise can cause a random displacement of each feature point of two sets related to the same subject, with a consequent lowering of the peak in the AM. In order to make the algorithm noise tolerant, the way to add the contribution in the AM has been changed: instead of adding a single contribution at the calculated coordinates, we add contribution in an area all around it. The area is user defined and it corresponds to a circular Roberts Window [10] which allows to choose how many points will carry contribution, with a fine step of variation. The area chosen for the Roberts Window gives a measure of the tolerance to noise.

When the AM is calculated, a similarity score has been extracted. Two identical features sets produce an AM with an absolute maximum value placed in the middle and various multiple local maxima of lower value. Local maxima usually have much lower values than the peak, but their difference depends on the Roberts Window chosen for the accumulation process. Differences in the two sets cause a decrease of the absolute maximum and an increase of the other values. Sets of points with no correlation can also produce multiple absolute maxima and multiple local maxima of high value (Fig. 8 (b)). We use this behaviour to extract a similarity score. Our algorithm can be described as follows:

- Calculate the absolute maximum value of the accumulation matrix;
- Calculate the gravitational center (C) of all absolute maxima positions (in case of high similarity, it will be coincident with the singular absolute maximum position);
- Calculate the mean value (HV) inside the Roberts Window centered in C;
- Calculate the mean value (LV) outside the Roberts Window centered in C;
- Normalize value LV: $LV' = LV/HV$;

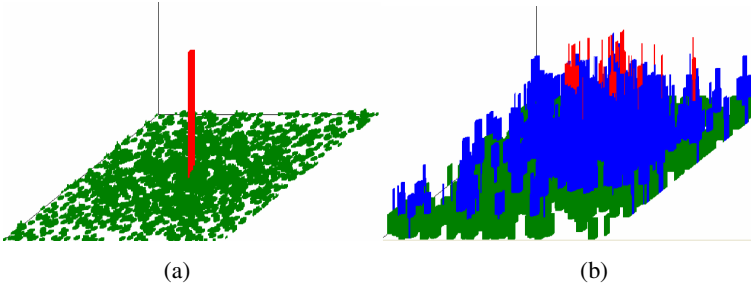


Fig. 8. 3D representation of the Accumulation Matrix. (a) Case of identity between clouds of points; (b) Case of clouds of points with no correlation.

The Roberts Window used for the mean calculation has the same order of that used for the accumulation process. The identified similarity score is proportional to the difference between values HV and LV. Since HV and LV depend also on the Roberts Windows, their difference have been related with the value obtained in case of perfect coincidence. Hence:

$$\text{Similarity score} = (1 - LV') / (1 - LV_i') \tag{3}$$

Where LV' is the normalized mean value outside the Roberts Window and LV_i' is the normalized mean value outside the Roberts Window calculated in case of identical features sets.

6 Experimental Results

The algorithm has been run on a database of 12 images, 512 x 512 pixel, of 10 different persons where two couple of images belongs to the same person. Figure 9 shows the mean trend of the similarity score relative to the order of the Roberts Window.

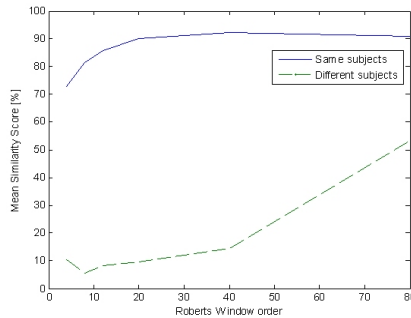


Fig. 9. Mean Similarity Score vs Roberts Window Order

Even though mean results reveal a good performance of the algorithm for a wide range of the order of the Roberts Window, in tests, an order of 8, which corresponds to a Window comparable to a circle of radius 3 pixel, minimizes best the score for different subjects.

Experimental evidence shows also that a threshold equal to 70% would bring both False Acceptance Rate and False Rejection Rate to a value of 0%.

7 Conclusions

In this paper is described a biometric system for personal identification, realized through the manipulation of retinal fundus images and the detection of its bifurcation points. The process consists of three phases: at first each retinal image is handled by pre-processing operations aiming to reduce its intrinsic noise; than, blood vessels are extracted to allow, subsequently, the detection of the bifurcation points necessary to personal identification. The satisfactory results of experimental encourage the authors to continue the studies in this direction.

References

1. Jain, A., Bolle, R., Pankanti, S.: *Biometrics: Personal Identification in Networked Society*, p. 123. Springer Science/Business Media, New York (1996)
2. Wilson, C.L., McCabe, R.M.: *Simple Test Procedure for Image-based Biometric Verification Systems*. NISTIR 6336, National Institute of Standards and Technology (1999), <http://www.itl.nist.gov/iaui/894.03/pubs.htm>
3. Tower, P.: *The Fundus Oculi in Monozygotic Twins: Report of Six Pairs of Identical Twins*. *Archives of Ophthalmology* 54, 225–239 (1955)
4. Simon, C., Goldstein, I.: *A New Scientific Method of Identification*. *New York State Journal of Medicine* 35(18), 901–906 (1935)
5. Kresimir, D.I., Mislav, G.: *A Survey of Biometric Recognition Methods*. In: 46th International SyrnPoSium Electronics in Marine, ELMAR-2004, Zadar, Croatia (2004)
6. Naka, K.I., Rushton, W.A.: *S-potentials from Luminosity Units in the Retina of fish (Cyprinidae)*. *Journal of Physiology* 185, 587–599 (1966)
7. Bevilacqua, V., Cariello, L., Cambò, S., Daleno, D., Mastronardi, G.: *Retinal Fundus Features Hybrid Detection based on a Genetic Algorithm*. In: *Proceedings of the Workshop on Medical Applications of Genetic and Evolutionary Computation (MedGEC)*, Seattle (USA) (2005)
8. Steger, C.: *An Unbiased Detector of Curvilinear Structures*. *Proceedings of IEEE Transactions on Pattern Analysis and Machine Intelligence* 20(2) (1998)
9. Ballard, D.H.: *Generalizing the Hough Transform to detect arbitrary shapes*. *Pattern Recognition* 13, 111–122 (1981)
10. Mastronardi, G., Daleno, D., Bevilacqua, V., Chiaia, G.: *Tecniche di identificazione personale basate sulla trasformata generalizzata di Hough applicata a nuvole di punti*. In: *Proceedings of National Conf. AICA 2007 (Associazione italiana per l'informatica ed il calcolo distribuito)*, Milano (Italy) (2007) ISBN 88-901620-3-1

Statistical Character Study of the Inert Coordinate Errors Caused by the Measurement Noise of the Netting Radar*

Wen-Bo Zhao¹, Xiao-Chun Mei², Chuan Li², and Yong-Ke Chen¹

¹ Artillery Academy of People Liberation Army, Heifei, Anhui, 230031, P.R. China

² East China Electronics Engineering Institute

Abstract. Considering the affection of measurement noise of the netting radar, the error components of the inert coordinates, including the netting radar inert coordinates and the netting fusion center inert coordinates, are given detail analysis in this paper. In the two kinds of the inert coordinates the corresponding statistical character of the errors caused by the measurement noise is educed in the form of the first order moment and the second order center moment. Lastly some conclusions are remarked.

1 Introduction

Compared with the isolated radar, the radar netting system [1,2,3] has many advantages, such as expanding the measurement space & time scope, enhancing the measurement precision and detection performance, greatly improving measured data transferring speed, and so on. Therefore the radar netting system would be widely used in future battles. The radar netting system consists of the netting fusion center, the netting radars and the communication chains. The netting radars measure the target in the airspace and export the corresponding plots to the netting fusion center by the communication chains. In the netting fusion center, the reported plots, firstly, will be transformed into the netting fusion center coordinates, then, will be applied for tracking the status of the target according to some data fusion algorithms. Generally it is the netting fusion center where the target track processing of the radar netting system is fulfilled. But measured information of the target in the airspace, such as range, bearing angle, elevation angle, is measured and extracted in the radar polar coordinates. In the netting fusion center, the plots data measured by the netting radar, must be transformed into the united coordinates in the netting fusion center by the some coordinate transformation technology [4,5,6,7], then can be used to the target status estimation. The data derivations of the radar netting system are the netting radars, which have different work modes, different kinds and different performance. With no considering the fixed errors of the netting radar and the

* This work was supported by the Natural Science Foundation of Anhui Province of China under grant No. 070412040.

coordinate transformation errors, the netting radar measure noise errors will lead to the different error demonstration to some extent in the radar inert coordinates and in the fusion center inert coordinates. The statistical performance of these noised data will affect target track quality. For example, before the Kalman filtering algorithm is used to track the target, the measure noise covariance matrix must be given. The purpose of the paper discuss the statistic performance of errors caused by the netting radar measurement noise in the two kinds of the inert coordinates.

2 In the Netting Radar Inert Coordinates the Statistic Performance Analysis of the Errors Caused by the Measurement Noise

2.1 The Coordinate Transformation Technology from the Polar Coordinates to the Netting Radar Inert Coordinates

Suppose that in time k the typical kinematics measured for a target in the airspace are range, bearing angle and elevation angle, represented by the vector, $[r_k, \theta_k, \varphi_k]^T$, in the netting radar polar coordinates. For the same target and in the same measuring time k , the vector, $[x_r^k, y_r^k, z_r^k]^T$, represents the inert coordinate of the netting radar, and the vector, $[x_f^k, y_f^k, z_f^k]^T$, denotes the inert coordinate of the netting fusion center. According to Reference [8], transforming the netting radar polar coordinate into the netting radar inert coordinate is given by

$$\begin{cases} x_r^k = r_k \cos \varphi_k \sin \theta_k \\ y_r^k = r_k \cos \varphi_k \cos \theta_k \\ z_r^k = r_k \sin \varphi_k \end{cases} \tag{1}$$

2.2 The Analysis of the Netting Radar Inert Coordinate Errors Caused by the Measurement Noise

From (1), it can be seen that the inert coordinate components of the netting radar can be expressed as the nonlinearity combination of measured components. Therefore, the error character of the netting radar inert coordinates must be tightly relevant to the statistic character of the measurement noise of the netting radar. According to (1), the inert coordinate error in the netting radar inert coordinates, expressed by the full differential linearity combination of the range error, bearing error and elevation angle error of the netting radar is given by

$$\begin{cases} dx_r^k = \frac{\partial x_r^k}{\partial r_k} dr_k + \frac{\partial x_r^k}{\partial \theta_k} d\theta_k + \frac{\partial x_r^k}{\partial \varphi_k} d\varphi_k \\ dy_r^k = \frac{\partial y_r^k}{\partial r_k} dr_k + \frac{\partial y_r^k}{\partial \theta_k} d\theta_k + \frac{\partial y_r^k}{\partial \varphi_k} d\varphi_k \\ dz_r^k = \frac{\partial z_r^k}{\partial r_k} dr_k + \frac{\partial z_r^k}{\partial \theta_k} d\theta_k + \frac{\partial z_r^k}{\partial \varphi_k} d\varphi_k \end{cases} \tag{2}$$

where dx_r^k, dy_r^k and dz_r^k represents the three error components of the netting radar inert coordinates. According to (1), the partial differential expresses of (2) can be representatively given by

$$\begin{cases} \frac{\partial x_r^k}{\partial r_k} = \cos \varphi_k \sin \theta_k, \frac{\partial x_r^k}{\partial \theta_k} = r_k \cos \varphi_k \cos \theta_k, \frac{\partial x_r^k}{\partial \varphi_k} = -r_k \sin \varphi_k \sin \theta_k \\ \frac{\partial y_r^k}{\partial r_k} = \cos \varphi_k \cos \theta_k, \frac{\partial y_r^k}{\partial \theta_k} = -r_k \cos \varphi_k \sin \theta_k, \frac{\partial y_r^k}{\partial \varphi_k} = -r_k \sin \varphi_k \cos \theta_k \\ \frac{\partial z_r^k}{\partial r_k} = \sin \varphi_k, \frac{\partial z_r^k}{\partial \theta_k} = 0, \frac{\partial z_r^k}{\partial \varphi_k} = r_k \cos \varphi_k \end{cases} \quad (3)$$

Inserting (3) into (2), the inert coordinate errors caused by the measurement noise can be derived as

$$\begin{cases} dx_r^k = \cos \varphi_k \sin \theta_k dr_k + r_k \cos \varphi_k \cos \theta_k d\theta_k - r_k \sin \varphi_k \sin \theta_k d\varphi_k \\ dy_r^k = \cos \varphi_k \cos \theta_k dr_k - r_k \cos \varphi_k \sin \theta_k d\theta_k - r_k \sin \varphi_k \cos \theta_k d\varphi_k \\ dz_r^k = \sin \varphi_k dr_k + r_k \cos \varphi_k d\varphi_k \end{cases} \quad (4)$$

2.3 In the Netting Radar Inert Coordinates the Statistic Character Analysis of Errors Caused by the Measurement Noise

The expectation of both sides of (4) can be determined as

$$\begin{cases} E [dx_r^k] = \cos \varphi_k \sin \theta_k E [dr_k] + r_k \cos \varphi_k \cos \theta_k E [d\theta_k] - r_k \sin \varphi_k \sin \theta_k E [d\varphi_k] \\ E [dy_r^k] = \cos \varphi_k \cos \theta_k E [dr_k] - r_k \cos \varphi_k \sin \theta_k E [d\theta_k] - r_k \sin \varphi_k \cos \theta_k E [d\varphi_k] \\ E [dz_r^k] = \sin \varphi_k E [dr_k] + r_k \cos \varphi_k E [d\varphi_k] \end{cases} \quad (5)$$

where $E[\cdot]$ denotes solving operation of mathematic expectation for the component of the bracket. Generally measuring range procedure, measuring bearing angle procedure and measuring elevation angle procedure of the netting radar are mutually independent, that is, every measurement error, $dr_k, d\theta_k, d\varphi_k$ is independent, zero mean, white Gaussian noise with variance $\delta_{r_k}^2, \delta_{\theta_k}^2, \delta_{\varphi_k}^2$. Therefore, the statistical character of measurement noise can be determined as

$$\begin{cases} E [dr_k] = 0 \\ E [d\theta_k] = 0 \\ E [d\varphi_k] = 0 \end{cases} \quad (6)$$

$$\begin{cases} E [(dr_k - E [dr_k])^2] = E [(dr_k)^2] = \delta_{r_k}^2 \\ E [(d\theta_k - E [d\theta_k])^2] = E [(d\theta_k)^2] = \delta_{\theta_k}^2 \\ E [(d\varphi_k - E [d\varphi_k])^2] = E [(d\varphi_k)^2] = \delta_{\varphi_k}^2 \end{cases} \quad (7)$$

$$\begin{cases} E [dr_k d\theta_k] = E [dr_k] E [d\theta_k] = 0 \\ E [d\theta_k d\varphi_k] = E [d\theta_k] E [d\varphi_k] = 0 \\ E [dr_k d\varphi_k] = E [dr_k] E [d\varphi_k] = 0 \end{cases} \quad (8)$$

Inserting (6) into (5), it can be derived as

$$\begin{cases} E [dx_r^k] = 0 \\ E [dy_r^k] = 0 \\ E [dz_r^k] = 0 \end{cases} \quad (9)$$

Considering the second order center moment of x -component of the netting radar inert coordinates, it can be defined as

$$\left\{ E \left[(dx_r^k - E [dx_r^k])^2 \right] = E \left[(dx_r^k)^2 \right] \right. \quad (10)$$

Inserting dx_r^k of (4) into (10), the second order center moment, , can be written as

$$E \left[(dx_r^k)^2 \right] = (\cos \varphi_k \sin \theta_k)^2 \delta_{r_k}^2 + (r_k \cos \varphi_k \cos \theta_k)^2 \delta_{\theta_k}^2 + (r_k \sin \varphi_k \sin \theta_k)^2 \delta_{\varphi_k}^2 \quad (11)$$

In the same manner of (11), the second order center moments of the y -component and z -component of the netting radar coordinates are derived as

$$E \left[(dy_r^k - E [dy_r^k])^2 \right] = E \left[(dy_r^k)^2 \right] = (\cos \varphi_k \cos \theta_k)^2 \delta_{r_k}^2 + (r_k \cos \varphi_k \sin \theta_k)^2 \delta_{\theta_k}^2 + (r_k \sin \varphi_k \cos \theta_k)^2 \delta_{\varphi_k}^2 \quad (12)$$

$$E \left[(dz_r^k - E [dz_r^k])^2 \right] = E \left[(dz_r^k)^2 \right] = (\sin \varphi_k)^2 \delta_{r_k}^2 + (r_k \cos \varphi_k)^2 \delta_{\varphi_k}^2 \quad (13)$$

The mutual covariance between x -component and y -component of the netting radar inert coordinates is expressed as

$$E \left[(dx_r^k - E [dx_r^k]) (dy_r^k - E [dy_r^k]) \right] = E [dx_r^k dy_r^k] = \sin \theta_k \cos \theta_k \left((\cos \varphi_k)^2 \delta_{r_k}^2 - (r_k)^2 (\cos \varphi_k)^2 \delta_{\theta_k}^2 + (r_k)^2 (\sin \varphi_k)^2 \delta_{\varphi_k}^2 \right) \quad (14)$$

In the same manner of (14) the mutual covariance between x -component and z -component, the mutual covariance between y -component and z -component, can be determined by (15) and (16).

$$E \left[(dx_r^k - E [dx_r^k]) (dz_r^k - E [dz_r^k]) \right] = E [dx_r^k dz_r^k] = \sin \theta_k \sin \varphi_k \cos \varphi_k \left(\delta_{r_k}^2 - (r_k)^2 \delta_{\varphi_k}^2 \right) \quad (15)$$

$$E \left[(dy_r^k - E [dy_r^k]) (dz_r^k - E [dz_r^k]) \right] = E [dy_r^k dz_r^k] = \cos \theta_k \sin \varphi_k \cos \varphi_k \left(\delta_{r_k}^2 - (r_k)^2 \delta_{\varphi_k}^2 \right) \quad (16)$$

2.4 Some Conclusions

According to the above analysis, some conclusions can be summarized as the followings.

(A) In the netting radar inert coordinates, the errors caused by the measurement noise, are not only relevant to the statistic character of measurement noise, but also relevant to measured values of the target. And the errors can be expressed as the nonlinearity combination of the statistic parameters of the measurement noise and measured values of the target by the netting radar.

(B) If the measurement noises are independent, zeros mean, white Gauss noises, the first order statistic character of the errors, caused by measurement noise, is also zeros mean.

(C) The second order statistic character of the inert coordinate errors, caused by measurement noise, is relevant to the measured values of the target and measurement variance.

(D) The measurement noise of the netting radar leads to the mutually coupling among the three components in the netting radar inert coordinates. Therefore, in order to no losing the coupling relation, while tracking targets, the method of the isolated single-dimension status filtering does not suggest to be used.

3 The Statistic Character Analysis of the Netting Fusion Center Inert Coordinate Errors Caused by Measurement Noise

3.1 The Coordinate Transform Technology from the Netting Radar Inert Coordinates to the Netting Fusion Center Inert Coordinates

According Reference [9], at time k , for the same target in the airspace, transforming the netting radar inert coordinate into the netting fusion center inert coordinate is given by

$$\begin{bmatrix} x_f^k \\ y_f^k \\ z_f^k \end{bmatrix} = (M_{ef}M_{er}^T) \begin{bmatrix} x_r^k \\ y_r^k \\ z_r^k \end{bmatrix} + \begin{bmatrix} x_f^{rk} \\ y_f^{rk} \\ z_f^{rk} \end{bmatrix} \tag{17}$$

where $[x_f^k, y_f^k, z_f^k]^T$ and $[x_f^{rk}, y_f^{rk}, z_f^{rk}]^T$ represent the netting fusion center inert coordinate of the target at time k and netting fusion center inert coordinate of the netting radar, where M_{ef} and M_{er} is representatively at the place of the netting radar the rotation matrix from the geocentric inert coordinates to the netting fusion center inert coordinates, and the rotate matrix from the geocentric inert coordinates to the netting radar inert coordinate, where $(M_{ef}M_{er}^T)^k$ enotes the rotation matrix from the netting radar inert coordinates to the netting fusion center inert coordinates. The rotation matrix M_{ef} and M_{er} is determined [9] by

$$M_{ef} = \begin{bmatrix} -\sin(L_f) & \cos(L_f) & 0 \\ -\sin(B_f)\cos(L_f) & -\sin(B_f)\sin(L_f)\cos(B_f) \\ \cos(B_f)\cos(L_f) & \cos(B_f)\sin(L_f)\sin(B_f) \end{bmatrix} \tag{18}$$

$$M_{er} = \begin{bmatrix} -\sin(L_r) & \cos(L_r) & 0 \\ -\sin(B_r)\cos(L_r) & -\sin(B_r)\sin(L_r)\cos(B_r) \\ \cos(B_r)\cos(L_r) & \cos(B_r)\sin(L_r)\sin(B_r) \end{bmatrix} \tag{19}$$

where L_f and B_f representatively denotes the earth longitude and the earth latitude of the netting fusion center, where L_r and B_r is representatively the earth longitude and the earth latitude of the netting radar. According to Reference [9], M_{ef} , M_{er} and $(M_{ef}M_{er}^T)^k$ are all orthogonal matrixes, and $(M_{ef}M_{er}^T)^k$ is

a symmetry orthogonal matrix. Considering symmetry matrix $(M_{ef}M_{er}^T)^k$, it is expressed as

$$(M_{ef}M_{er}^T)^k = \begin{bmatrix} m_{11}^k & m_{12}^k & m_{13}^k \\ m_{12}^k & m_{22}^k & m_{23}^k \\ m_{13}^k & m_{23}^k & m_{33}^k \end{bmatrix} \tag{20}$$

Inserting (20) into (17), it can be derived by

$$\begin{cases} x_f^k = m_{11}^k x_r^k + m_{12}^k y_r^k + m_{13}^k z_r^k + x_f^{rk} \\ y_f^k = m_{12}^k x_r^k + m_{22}^k y_r^k + m_{23}^k z_r^k + y_f^{rk} \\ z_f^k = m_{13}^k x_r^k + m_{23}^k y_r^k + m_{33}^k z_r^k + z_f^{rk} \end{cases} \tag{21}$$

3.2 The Analysis of the Netting Fusion Center Inert Coordinate Error Caused by the Measurement Noise

At time k , according to (17) ~ (21), it is known that $[x_f^{rk}, y_f^{rk}, z_f^{rk}]^T$ and $(M_{ef}M_{er}^T)^k$ are the fixed vector and the fixed matrix. By variant of the netting radar inert coordinate of the given target, the full differential express of the netting fusion center inert coordinate errors is determined as

$$\begin{cases} dx_f^k = \frac{\partial x_f^k}{\partial x_r^k} dx_r^k + \frac{\partial x_f^k}{\partial y_r^k} dy_r^k + \frac{\partial x_f^k}{\partial z_r^k} dz_r^k = m_{11}^k dx_r^k + m_{12}^k dy_r^k + m_{13}^k dz_r^k \\ dy_f^k = \frac{\partial y_f^k}{\partial x_r^k} dx_r^k + \frac{\partial y_f^k}{\partial y_r^k} dy_r^k + \frac{\partial y_f^k}{\partial z_r^k} dz_r^k = m_{12}^k dx_r^k + m_{22}^k dy_r^k + m_{23}^k dz_r^k \\ dz_f^k = \frac{\partial z_f^k}{\partial x_r^k} dx_r^k + \frac{\partial z_f^k}{\partial y_r^k} dy_r^k + \frac{\partial z_f^k}{\partial z_r^k} dz_r^k = m_{13}^k dx_r^k + m_{23}^k dy_r^k + m_{33}^k dz_r^k \end{cases} \tag{22}$$

3.3 The Statistical Character Analysis of the Netting Fusion Center Inert Coordinate Errors Caused by the Measurement Noise

According to (9) and (22), the mathematic expectation of the netting fusion center inert coordinate errors caused by the measurement noise can be expressed as

$$\begin{cases} E \left[dx_f^k \right] = m_{11}^k E \left[dx_r^k \right] + m_{12}^k E \left[dy_r^k \right] + m_{13}^k E \left[dz_r^k \right] = 0 \\ E \left[dy_f^k \right] = m_{12}^k E \left[dx_r^k \right] + m_{22}^k E \left[dy_r^k \right] + m_{23}^k E \left[dz_r^k \right] = 0 \\ E \left[dz_f^k \right] = m_{13}^k E \left[dx_r^k \right] + m_{23}^k E \left[dy_r^k \right] + m_{33}^k E \left[dz_r^k \right] = 0 \end{cases} \tag{23}$$

The second order center moment of the x -component of the netting fusion center inert coordinates is determined by

$$\begin{aligned} E \left[\left(dx_f^k - E \left[dx_f^k \right] \right)^2 \right] &= E \left[\left(dx_f^k \right)^2 \right] \\ &= (m_{11}^k)^2 E \left[\left(dx_r^k \right)^2 \right] + (m_{12}^k)^2 E \left[\left(dy_r^k \right)^2 \right] + (m_{13}^k)^2 E \left[\left(dz_r^k \right)^2 \right] \\ &\quad + 2m_{11}^k m_{12}^k E \left[dx_r^k dy_r^k \right] + 2m_{12}^k m_{13}^k E \left[dy_r^k dz_r^k \right] + 2m_{11}^k m_{13}^k E \left[dx_r^k dz_r^k \right] \end{aligned} \tag{24}$$

In the same manner of (24), the second order center moments of the y -component and z -component of the netting fusion center inert coordinates can be representatively expressed as (25) and (26).

$$\begin{aligned}
 E \left[\left(dy_f^k - E \left[dy_f^k \right] \right)^2 \right] &= E \left[\left(dy_r^k \right)^2 \right] \\
 &= \left(m_{12}^k \right)^2 E \left[\left(dx_r^k \right)^2 \right] + \left(m_{22}^k \right)^2 E \left[\left(dy_r^k \right)^2 \right] + \left(m_{23}^k \right)^2 E \left[\left(dz_r^k \right)^2 \right] \\
 &+ 2m_{12}^k m_{22}^k E \left[dx_r^k dy_r^k \right] + 2m_{22}^k m_{23}^k E \left[dy_r^k dz_r^k \right] + 2m_{12}^k m_{23}^k E \left[dx_r^k dz_r^k \right]
 \end{aligned} \tag{25}$$

$$\begin{aligned}
 E \left[\left(dz_f^k - E \left[dz_f^k \right] \right)^2 \right] &= E \left[\left(dz_r^k \right)^2 \right] \\
 &= \left(m_{13}^k \right)^2 E \left[\left(dx_r^k \right)^2 \right] + \left(m_{23}^k \right)^2 E \left[\left(dy_r^k \right)^2 \right] + \left(m_{33}^k \right)^2 E \left[\left(dz_r^k \right)^2 \right] \\
 &+ 2m_{13}^k m_{23}^k E \left[dx_r^k dy_r^k \right] + 2m_{23}^k m_{33}^k E \left[dy_r^k dz_r^k \right] + 2m_{13}^k m_{33}^k E \left[dx_r^k dz_r^k \right]
 \end{aligned} \tag{26}$$

The mutual covariance between the x -component and y -component of the netting fusion center inert coordinates is determined by

$$\begin{aligned}
 E \left[\left(dx_f^k - E \left[dx_f^k \right] \right) \left(dy_f^k - E \left[dy_f^k \right] \right) \right] &= E \left[dx_f^k dy_f^k \right] \\
 &= m_{11}^k m_{12}^k E \left[\left(dx_r^k \right)^2 \right] + m_{12}^k m_{22}^k E \left[\left(dy_r^k \right)^2 \right] + m_{13}^k m_{23}^k E \left[\left(dz_r^k \right)^2 \right] \\
 &+ E \left[dx_r^k dy_r^k \right] \left(m_{11}^k m_{22}^k + \left(m_{12}^k \right)^2 \right) \\
 &+ E \left[dx_r^k dz_r^k \right] \left(m_{11}^k m_{23}^k + m_{12}^k m_{13}^k \right) \\
 &+ E \left[dy_r^k dz_r^k \right] \left(m_{12}^k m_{23}^k + m_{13}^k m_{22}^k \right)
 \end{aligned} \tag{27}$$

In the same manner of (27), the mutual covariance between the x -component and z -component, and the mutual covariance between y -component and z -component of the netting fusion center inert coordinates are written by (28) and (29).

$$\begin{aligned}
 E \left[\left(dx_f^k - E \left[dx_f^k \right] \right) \left(dz_f^k - E \left[dz_f^k \right] \right) \right] &= E \left[dx_f^k dz_f^k \right] \\
 &= m_{11}^k m_{13}^k E \left[\left(dx_r^k \right)^2 \right] + m_{12}^k m_{23}^k E \left[\left(dy_r^k \right)^2 \right] + m_{13}^k m_{33}^k E \left[\left(dz_r^k \right)^2 \right] \\
 &+ E \left[dx_r^k dy_r^k \right] \left(m_{11}^k m_{23}^k + m_{12}^k m_{13}^k \right) \\
 &+ E \left[dx_r^k dz_r^k \right] \left(m_{11}^k m_{33}^k + \left(m_{13}^k \right)^2 \right) \\
 &+ E \left[dy_r^k dz_r^k \right] \left(m_{12}^k m_{33}^k + m_{13}^k m_{23}^k \right)
 \end{aligned} \tag{28}$$

$$\begin{aligned}
 E \left[\left(dy_f^k - E \left[dy_f^k \right] \right) \left(dz_f^k - E \left[dz_f^k \right] \right) \right] &= E \left[dy_f^k dz_f^k \right] \\
 &= m_{12}^k m_{13}^k E \left[\left(dx_r^k \right)^2 \right] + m_{22}^k m_{23}^k E \left[\left(dy_r^k \right)^2 \right] + m_{23}^k m_{33}^k E \left[\left(dz_r^k \right)^2 \right] \\
 &+ E \left[dx_r^k dy_r^k \right] \left(m_{12}^k m_{23}^k + m_{13}^k m_{22}^k \right) \\
 &+ E \left[dx_r^k dz_r^k \right] \left(m_{12}^k m_{33}^k + m_{13}^k m_{23}^k \right) \\
 &+ E \left[dy_r^k dz_r^k \right] \left(m_{22}^k m_{33}^k + \left(m_{23}^k \right)^2 \right)
 \end{aligned} \tag{29}$$

Inserting $E[(dx_r^k)^2], E[(dy_r^k)^2], E[(dz_r^k)^2], E[dx_r^k dy_r^k], E[dx_r^k dz_r^k]$ and $E[dx_r^k dz_r^k]$ of (11)~(16) into (24)~(29), the second order center moments of the netting fusion inert coordinate errors can be solved.

3.4 Some Conclusions

Based on the above analysis, some conclusions can be summarized as the followings.

(A) The netting fusion center inert coordinate errors, caused by the measurement noise, can be described as the linearity function of the three components of the netting radar inert coordinate error.

(B) The mean of the netting fusion center inert coordinate error, caused by measurement noise, is zeros, because the mean of the netting radar inert coordinate error is also zero mean.

(C) The second order center moments of the netting fusion center inert coordinate error is linearity combination of the second order center moments of the netting radars inert coordinate errors.

(D) In the netting fusion center inert coordinates, the error components, caused by the measurement noise, are mutually coupling. While tracking the targets in the inert coordinates, the coupling relation must be considered, otherwise, the tracking performance will be debased.

References

1. Li, N.J.: The Efficiency Evaluation Study of Surveillance Radars Networks. Modern Radar 2 (2000)
2. Shao, X.J., Zhou, L.: The Radar Netting Technology Study of the Early Warning Detection System. Modern Radar 9 (2003)
3. Xie, H., Yang, Z.Q., He, F.: The Modeling Study of Anti-jamming Ability of the Ground Surveillance Radars Networks. The Technology of Electron Confrontation 18(2) (2001)
4. Farina, M.A., Studer, F.A.: Radar Data Processing. vol. I, II, Research Studies Press LTD (1985)
5. Johson, G.W.: Choice of Coordinates and Computational Difficulty. IEEE Trans. On AC 19(2) (1974)
6. Cantrel, B.H., Grindlay, A., Dodage, C.H.: Formulation of a Platform-to-Platform Integration System. NRL Memorandum Report, vol. 3404 (1976)
7. Ted, J., Broida: Choice of Coordinate System for Multiple Sensor Fusion. In: SPIE, vol. 1611, Sensor Fusion IV (1991)
8. Ceng, H.W., Sun, H.K.: On the Influence of Coordinate Transform upon Measurement Error of Long-baseline Distributed Sensors System. In: SPIE 1997, vol. 3067, pp. 136-145 (1997)
9. He, Y., et al.: Radar Data Process and Application, pp. 66-67. Publishing House of Electronic Industry (January 2006)

Author Index

- Abdel-Magid, Youssef Lotfy 684
Abido, Mohammad Ali 684
Adachi, Hirotaka 675
Adam, Sébastien 430
Anagnostou, Miltiades 27
Armenise, Mario Nicola 1039
Azzariti, Amalia 1085
- Bai, Bendu 566
Bai, Xuejie 308
Bai, Zhifeng 468
Ben, Shenglan 622
Berardi, Francesco 1085
Bernard, Simon 430
Bevilacqua, Vitoantonio 1039, 1077,
1085, 1142, 1166, 1200, 1210, 1229
- Cai, Jianyong 76
Çakır, O. 347
Calabrese, Marco 896, 1192
Cao, Binggang 468
Cariello, Lucia 1077, 1166, 1229
Carnimeo, Leonarda 1069, 1175
Casorio, Pasquale 1200
Castellano, Marcello 1077, 1229
Chang, Pei-Chann 292
Chang, Ruei-Yuan 758
Chang, Wenbing 922
Chau, Kwok-Wing 583
Che, Chunhui 449
Chen, Deqiang 209
Chen, Hanning 541
Chen, Ke 68, 807
Chen, Shih-Hsin 292
Chen, Shuhong 300
Chen, Wang 140, 742
Chen, Weiwei 99
Chen, Yi-Ping Phoebe 888
Chen, Yong-Ke 1127, 1238
Chen, Yu 935
Chen, Yu-Wang 251
Ciccimarra, Alessandro 1142
Ciminelli, Caterina 1039
Colabufo, Nicola A. 1085
Columbo, Donatello 1229
- Daleno, Domenico 1229
Decataldo, Giacinto 1077
Dellisanti Fabiano, Massimiliano 1229
Di Bernardo, Diego 1085
Di Lecce, Vincenzo 896, 1192
Ding, Shifei 661, 701
Dong, Dong 410
Du, Ji-Yan 1127
Du, Yajun 840
- Endo, Satoshi 225
Erig Lima, Carlos R. 959, 1109
Esin, Yesim H. 558
- Fan, Jiu-Lun 484
Fan, Zhongshan 848
Feng, Enmin 709
Feng, Yong 526
Filograno, Giuseppe 1210
Friedline, Christopher J. 1004
Fu, Jinming 799
- Gao, Chao 367
Gao, Li 717
Gao, Liang 40
Gao, Lixin 201
Gao, Weixin 616
García, Juan Carlos Figueroa 275, 316
Giannini, Marco 1166, 1229
Gol Alikhani, Mohsen 123
Gong, Dunwei 267
Guan, Haiwa 201
Guan, Yan-Yong 857
Gui, Jie 1055
Güner, A. 347
Guo, Chonghui 709
Guo, Guifang 468
Guo, Lanshen 107
Guo, Qinglin 259
Guo, Wanyou 717
- Hai, Yufeng 840
Han, Fei 654
Han, Jun 410

- Han, Leng 975
 Hao, Fang-Fang 308
 He, Haitao 449
 He, Xiaoxian 541
 Heath, Lenwood 1030
 Hekim, Mahmut 667
 Heutte, Laurent 430
 Ho, Anna 864
 Hu, Kunyuan 541
 Huang, De-Shuang 605
 Huang, Hui-Ming 1183
 Huang, Jian 182
 Huang, Liyu 717
 Huang, Qinhua 906
 Huang, Wenhui 996
 Huang, Yuanguai 717
 Huang, Zhi-Kai 1183

 Ibrikci, Turgay 667
 Iorio, Francesco 1085

 Javadian, Nikbakhsh 123
 Ji, Guangrong 693
 Jia, Weikuan 661, 701
 Jian, Chengfeng 824
 Jin, Chenxia 367
 Jin, Fengxiang 701
 Jin, Yaochu 235

 Kaku, Ikou 922
 Kalenatic, Dusko 275
 Kang, Lishan 164
 Kang, Zhou 164
 Kashki, Mohammad 684
 Kaya, I. 347
 Khorasani, K. 642
 Kim, Byeoung-su 1117
 Kim, ByungKi 84, 326
 Kim, Culhyun 84
 Kim, Jae-Jun 1117
 Kim, Whoi-Yul 1117
 Kim, Young-Ouk 476
 Kobayashi, Kunikazu 675
 Kumar, Th. Shanta 880
 Kuremoto, Takashi 675

 Lai, Guoming 633
 Lal, Sunil Pranit 225
 Lawson, Mark J. 1030
 Le, Minh Nghia 131
 Le, Van Hung 420, 438
 Lee, DongSu 326
 Lee, Guanling 758
 Lee, Gwang-Gook 1117
 Lee, Hong-Hee 190
 Lee, JaeHee 326
 Lee, MinTae 84
 Lee, SangHyun 84, 326
 Lee, Yang Weon 534
 Lei, Zhen 872
 Leone, Ilenia 1142
 Li, Bo 1047, 1093
 Li, Chuan 1238
 Li, Dong 872
 Li, Fachao 367
 Li, Gang 888
 Li, Li 156, 384, 494
 Li, Lin 616
 Li, LunBo 90
 Li, Shujuan 775
 Li, Tao 943
 Li, Wei 60
 Li, Xuelin 1101
 Li, Yan 775
 Li, Ying 566
 Li, Yiping 864
 Li, Zhan 807
 Liang, Yanjun 115
 Liao, Jianfeng 872
 Lim, Dudy 235
 Liu, Chunmei 1158
 Liu, Fei 420, 438
 Liu, Kun-Hong 340
 Liu, Yankui 308
 Liu, Yong 775
 Liu, Yuyang 84
 Liu, Zengrong 990
 Liu, Zhijing 832
 Lok, Tat-Ming 1012
 Long, Rong 799
 Lopes, Heitor S. 959, 1109
 Lopez Bello, Cesar Amilcar 275
 Lu, Xueyan 518
 Luo, Dehan 502
 Luo, Shuanghu 906
 Luo, Xiaonan 449
 Luo, Yan 332
 Lv, Pin 750
 Lyu, Michael R. 1012

- Ma, Feiteng 449
 Ma, GuangFu 90
 Ma, L. 642
 Maguire, Liam 76
 Majewski, Maciej 46
 Mao, Weihua 19
 Mastronardi, Giuseppe 1039, 1077,
 1085, 1142, 1166, 1200, 1210, 1229
 McGinnity, T.M. 76
 Mei, Liquan 597
 Mei, Xiao-Chun 1238
 Menolascina, Filippo 1039, 1085
 Miao, Duoqian 1158
 Moon, KyungIl 326
 Mu, Xiangyang 616

 Neruda, Roman 284
 Nguyen, Ngoc-Tu 190
 Nian, Rui 693
 Nie, Rencan 9
 Niu, Ben 156, 384, 494, 541

 Obayashi, Masanao 675
 Oliveira, Francisco 864
 Ong, Yew-Soon 131, 235
 Orhan, Umut 667
 Ouyang, Weimin 906
 Özen, A. 347

 Palmieri, Vito 1166
 Pan, Deng 967
 Pan, Zhenghua 401, 815
 Pannarale, Paolo 1085
 Papaioannou, Ioannis 27
 Paradiso, Angelo 1085
 Park, Chang-Woo 476
 Park, ChulSu 84, 326
 Park, Dong-Chul 376
 Park, Geum-Dal 174
 Park, SeungBeom 84, 326
 Peng, Sheng-Lung 758
 Perrone, Roberto 1085
 Pisciotta, Luca 1077

 Qi, Yanxia 107
 Qin, Renchao 943

 Raju, G. 880, 914
 Ramakrishnan, Naren 1030
 Rao, Shuqin 182

 Rong, Qiguo 1022
 Roussaki, Ioanna 27

 Sabu, M.K. 914
 Salum, Latif 783
 Schmidt, Sam 888
 Sendhoff, Bernhard 235
 Shang, Lihui 217
 Shen, Jianwei 990
 Shen, Yan 1221
 Shi, Ming-Guang 1012
 Shi, Yan-jun 140, 742
 Shi, Zhenwei 709
 Shi, Zhongzhi 701
 Shin, Sang-Ho 174
 Shuai, Dianxun 510, 549, 790
 Silva, Rafael R. 1109
 Slušný, Stanislav 284
 Soldo, Domenico 896
 Soriano Mendez, Jose Jairo 316
 Soysal, B. 347
 Stuart, Keith Douglas 46
 Su, Bing 975
 Su, Chunyang 661, 701
 Su, Guangda 622
 Su, Hong 54
 Sun, Jingyu 840
 Sun, Kai 251
 Sun, Zhaohao 410
 Supciller, Aliye Ayca 783

 Tagliaferri, Roberto 1085
 Tang, Lixin 148, 727
 Tang, Nan 616
 Tarricone, Gianfranco 1077
 Tavakkoli-Moghaddam, Reza 123
 Tavares, Leonardo G. 959
 Teng, Hong-fei 140, 742
 Thinley, Sangay 880
 Thomas, Binu 880
 Tommasi, Stefania 1085
 Tran, Dinh Khang 420, 438
 Tuğcu, E. 347

 Unel, Mustafa 558

 Valderrama-Gonzalez, G.D. 76
 Verleysen, Michel 693
 Vidnerová, Petra 284

- Wan, Anhua 19
 Wang, Cen 401
 Wang, Chao 1047
 Wang, Chunheng 1158
 Wang, Chunyang 1
 Wang, Hong-Qiang 1093
 Wang, Jianmin 449
 Wang, JianWei 1221
 Wang, Jiayao 848
 Wang, Jina 182
 Wang, Li-Ming 1127
 Wang, Shulin 1055, 1101
 Wang, Wenjuan 766
 Wang, Xianpeng 727
 Wang, Xiao-Feng 1135
 Wang, Xiao-Juan 40
 Wang, Xin 661
 Wang, Yi 990
 Wang, Yujia 575
 Wang, Zhongsheng 115
 Wei, XiaoPeng 99, 1221
 Wei, Zhi 107
 Wen, Guo-Zhu 605
 Wong, Fai 864
 Wu, Jie 1
 Wu, Kai-gui 526
 Wu, Lihui 591
 Wu, Lingda 872
 Wu, QingXiang 76
 Wu, Xiaoqian 209
 Wu, Yong 54
 Wu, Youshou 622
 Wu, Zhong-fu 526

 Xiang, Jian 1150
 Xiao, Xiuchun 60
 Xiao, Yiyong 922
 Xie, Min 201
 Xiong, Zenggang 243
 Xu, Chuanyu 459
 Xu, Chun-Gui 340
 Xu, Gang 468
 Xu, Huan 1135
 Xu, Peng 468
 Xu, Xinzheng 661
 Xu, Yuefei 775
 Xu, Yunpeng 391
 Xue, Shaoe 182
 Xue, Yang 951

 Yamada, Koji 225
 Yan, Guozheng 1
 Yan, Nin 115
 Yan, Zhuangzhi 935
 Yang, Fang 857
 Yang, Gen-Ke 251
 Yang, Guang 391
 Yang, Ming 164
 Yang, Ruolin 975
 Yang, Shengyun 633
 Yang, Wenlong 1
 Yang, XiaoFan 54
 Yang, Yang 243
 Yang, Yuekui 840
 Yang, Yupu 575
 Ye, Chun-xiao 526
 Ye, Xiaowang 832
 Yi, Chenfu 60, 807
 Yi, Chong Ho 476
 Yildiz, Mehmet 558
 Yin, Jian 182
 Yoneda, Kentaro 675
 Yoo, Kee-Young 174
 Yoon, Ja-Young 1117
 Yoshida, Yuji 358
 You, Zhuhong 1055
 Yu, Changrui 332
 Yu, Hai-Yan 484
 Yu, Lianzhi 1
 Yu, Zhao-Xia 857
 Yuan, Dehui 633
 Yuan, Fang 982
 Yuan, Jie 267
 Yuan, Lin 750
 Yuan, Xiaofeng 300
 Yue, Shuai 68

 Zan, Peng 1
 Zehner, Zendra E. 1004
 Zeng, Ming 243
 Zeng, Zhibin 775
 Zhang, Changshui 391
 Zhang, Chao-Yong 40
 Zhang, Dexian 848
 Zhang, Haiwei 107
 Zhang, Hongjuan 709
 Zhang, Jie 591
 Zhang, Jinfang 750
 Zhang, Jun 583
 Zhang, Lei 1093

- Zhang, Liqing 967, 996, 1030
Zhang, Liyan 799
Zhang, Ming 259
Zhang, Mingwang 734
Zhang, Qingfu 292
Zhang, Shanwen 1055, 1101
Zhang, Shujuan 597
Zhang, Xiaoxia 148
Zhang, Xuemin 243
Zhang, Xueping 848, 1004
Zhang, Ya Jing 951
Zhang, Yanning 566
Zhang, Yu 943
Zhang, Yunong 60, 68, 807
Zhang, Zhe 1022
Zhao, Dongfeng 9
Zhao, Tingting 99
Zhao, Wen-Bo 1127, 1238
Zhao, Yuqin 734
Zhao, Zhongming 975, 1004
Zheng, Chun-Hou 1093
Zheng, Jie 209
Zheng, Jing 391
Zheng, XiaoYang 54
Zheng, Yalin 391
Zhong, Jiang 526
Zhong, Tongke 60
Zhou, Dongming 9
Zhou, Jianqin 766
Zhou, Shiqiong 468
Zhou, Yanhong 982
Zhou, Yatong 616
Zhou, Yongquan 518
Zhu, Hongli 1150
Zhu, Qiong 591
Zhu, Tianqiao 717
Zhu, Yunlong 541
Zou, Yuhua 502